

# Retrieve, Annotate, Evaluate, Repeat: Leveraging Multimodal LLMs for Large-Scale Product Retrieval Evaluation

Kasra Hosseini, Thomas Kober, Josip Krapac, Roland Vollgraf,  
Weiwei Cheng, Ana Peleteiro Ramallo,

Zalando SE, Berlin, Germany

## Abstract

Evaluating production-level retrieval systems at scale is a crucial yet challenging task due to the limited availability of a large pool of well-trained human annotators. Large Language Models (LLMs) have the potential to address this scaling issue and offer a viable alternative to humans for the bulk of annotation tasks. In this paper, we propose a framework for assessing the product search engines in a large-scale e-commerce setting, leveraging Multimodal LLMs for (i) generating tailored annotation guidelines for individual queries, and (ii) conducting the subsequent annotation task. Our method, validated through deployment on a large e-commerce platform, demonstrates comparable quality to human annotations, significantly reduces time and cost, facilitates rapid problem discovery, and provides an effective solution for production-level quality control at scale.

## 1 Introduction

Search functionality is a fundamental component of e-commerce platforms, with the objective of finding the most relevant products in a dynamic product database. Customers using search often exhibit a higher intent to find specific products (Moe, 2003), leading to greater engagement and conversion rates. However, they may struggle to articulate their needs in a search query. Even if they do express their intent clearly, information retrieval (IR) systems might fail to interpret it correctly, resulting in irrelevant search results (Wang and Na, 2024).

Evaluating product retrieval systems on a large scale in a multilingual setting and for a diverse set of customer queries is an intricate but essential task for maintaining a high-quality user experience and driving business success. A prerequisite for this evaluation is the availability of a large enough pool of query-product relevance labels (Voorhees, 2001; Halvey et al., 2015), which indicate whether

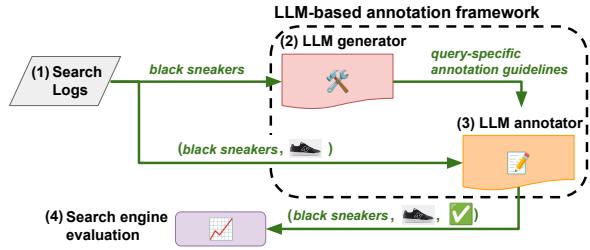


Figure 1: Our proposed framework works by extracting a query-product pair from our search query-click logs (1). The query (e.g. *black sneakers*) is then passed on to the **LLM generator** (2). The LLM generator creates *specific annotation instructions* for the given query. The query-specific annotation guidelines and the query-product pair (e.g. *black sneakers* and the retrieved product) are provided as input to the **LLM annotator** (3). Lastly, the annotated query-product pair is forwarded to the search engine evaluation module (4).

a retrieved product is semantically relevant to the query. Semantic relevance depends solely on the query and the product, excluding other contextual factors such as personal customer preferences.

Creating annotation guidelines that codify what is semantically relevant is a complex task (Spark-Jones, 1975). It requires describing the guidelines in a digestible, concise, yet precise manner, as well as curating a set of illustrative examples of varying difficulty. Even with well-defined guidelines and well-trained human annotators, manual annotation is slow and costly.

The advent of crowd-sourcing platforms has increased scalability (Blanco et al., 2011; Alonso and Mizzaro, 2012; Lease and Yilmaz, 2013; Marcus et al., 2015; Chen et al., 2016), allowing for a trade-off between speed and cost. However, increasing the number of annotators can lead to inconsistencies, as even the same annotator may provide contradictory annotations for the same query-product pair, let alone multiple annotators. Consistency can be improved by using more annotators per pair (see,

e.g., Ferrante et al. (2017)), but this results in increased cost. In large e-commerce systems, the volume of data that needs to be annotated leads to prohibitively high costs when using crowd-sourcing platforms that rely on human annotators.

While the rate of manual relevance judgement varies depending on the task (Voorhees, 2001; Sanderson et al., 2010; Chen et al., 2022; Soviero et al., 2024), in our use case, we estimate a throughput of 2-3 query-product pairs per minute. As an example, 50,000 queries<sup>1</sup> and 20 products per query results in one million query-product pairs, which takes 5,500-8,500 hours of human labour, assuming one annotation per pair. Moreover, evaluation is not a one-off practice; ideally, companies continuously assess their search engines to ensure effectiveness over time.

The sheer volume of required annotations in multiple languages, along with the need for continuous evaluation, makes human-generated relevance judgements the primary bottleneck in creating product retrieval evaluation datasets. To overcome these challenges, there has been growing interest in leveraging LLMs (Faggioli et al., 2023; Thomas et al., 2023; Soviero et al., 2024; Rahmani et al., 2024; Upadhyay et al., 2024; Bergum, 2024).

In this study, we propose a framework that leverages the capabilities of Multimodal Large Language Models (MLLMs) for assessing the relevance of query-product pairs (Fig. 1). Our method combines the strengths of LLMs and MLLMs in understanding natural language queries across various languages and processing both textual and visual features of products. Unlike traditional per-task annotation guidelines, such as those discussed by Soviero et al. (2024), we employ LLMs to generate annotation guidelines *specific to each query*. Additionally, our pipeline’s modular design allows for caching and parallel processing, which is crucial for scaling up to larger systems. This framework has enabled daily evaluations of our product retrieval systems. It has also facilitated the comparison of different search models, increasing our confidence in offline evaluations and complementing our online evaluation techniques, such as A/B testing and other controlled online experiments (Kohavi et al., 2009). Moreover, we have used the relevance assessments’ outputs to train, evaluate and analyse other components of our search and ranking systems.

<sup>1</sup>A modest number for the evaluation of large IR systems.

We furthermore show that while human-human and human-LLM agreement scores are on par with each other, we find that humans and LLMs tend to make very different types of annotation errors. Our findings suggest that LLMs are very effective for the bulk work of annotations whereas human expertise is better leveraged for more complex cases.

In summary, our contributions are as follows:

- We introduce a multimodal LLM-based evaluation framework for large-scale product retrieval systems. We propose *query-level* annotation guidelines and utilise multimodal inputs (text and images) for relevance assessment.
- We evaluate the performance of our framework against human annotations on real-world production search queries in a multilingual setting and analyse the different types of errors that humans and LLMs tend to make.
- We demonstrate the cost-effectiveness and efficiency of our approach for conducting large-scale evaluations. We also compare the performance of different types of LLMs (Radford et al., 2019; Brown et al., 2020; Achiam et al., 2023) for relevance assessment.

## 2 Multimodal LLM-based relevance assessment

The setup of our method is depicted in Fig. 2. It is designed to leverage the capabilities of (M)LLMs for efficient evaluation of large-scale product retrieval systems, and it consists of six main steps: (1) For a given query and its context (e.g., selected gender and market), an LLM generates a query requirement list and a *query-specific* annotation guideline. The query requirement list captures the relevant pieces of information in the user’s query and their level of importance. For example, for the query *Nike red shoes*, the query requirement list includes the brand (*Nike*), colour (*red*) and product category (*shoe*). The query-specific annotation guideline is generated by the LLM based on the query and its requirement list. It outlines criteria for each predefined label (see Appendix A for a detailed example). In our experiments, we defined three relevance labels for a query-product pair: “irrelevant”, “acceptable substitute” and “highly relevant”.<sup>2</sup>

<sup>2</sup>See Appendix B for details.

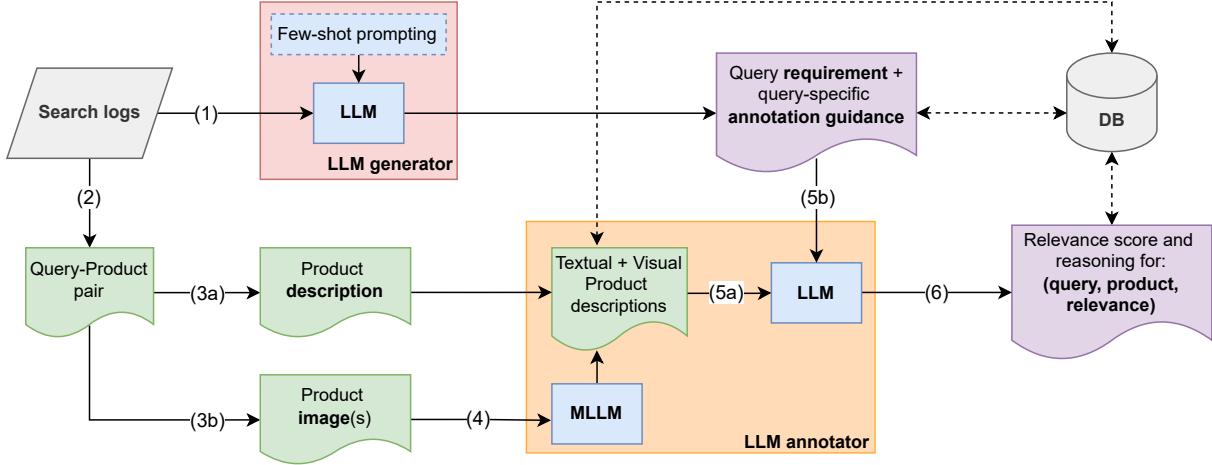


Figure 2: Our proposed Multimodal LLM-powered framework enables offline evaluation of large-scale product retrieval systems and presents significant time and cost reductions compared to existing evaluation techniques. Refer to Fig. 1 for an overview of the main steps in the framework, and consult the text for further details. The orange rectangle indicates where a “one-step” Multimodal LLM (MLLM) could be utilised, instead of employing one MLLM to create a textual description for image inputs (Step 4) followed by an LLM (Step 5a). In the one-step MLLM, both textual descriptions and the product image are directly fed into the LLM annotator, along with query requirements and query-specific annotation guidelines. The depiction of the pipeline is simplified for readability.

(2) The query and its context are sent to the search engine, which retrieves a set of products. For simplicity, we illustrate this process using a single query-product pair. However, in practice, we work with multiple query-product pairs and may utilise two or more retrieval systems, particularly when comparing their performance.

(3a,b) For each retrieved product, we have access to its textual description and its associated image.

(4) Using MLLMs and the product image, a visual description in textual form is generated.

(5a,b) The combined textual and visual product descriptions are sent to an LLM together with the outputs of Step 1 (i.e., query requirement list and query-specific annotation guideline).

(6) The LLM assigns a relevance score to the query-product pair using a set of predefined labels. In its simplest form, the output is a database with one row for each (query, product, relevance score).

In Steps 1 and 6, we utilise chain-of-thought (CoT) prompting (Wei et al., 2022; Nye et al., 2021) to enhance the quality of (M)LLM outputs and for debugging. An example of the reasoning steps is shown in Appendix A.

As illustrated with dashed lines in Fig. 2, all outputs and intermediate steps are stored in a database. This caching serves two key functions in our pipeline. Firstly, it facilitates efficient retrieval and reuse. When evaluating a new search engine

configuration (or a variation of existing ones), the database is queried to retrieve relevant pieces of information, including the query requirement list, query-specific annotation guidance, textual and visual product descriptions, and relevance scores. We only compute the missing pieces of information. Secondly, it ensures consistent evaluation across different search engines, as intermediate steps (such as query-specific annotation guidelines) are computed only once and then used to evaluate various search engines.

### 3 Experiments and Results

#### 3.1 Dataset

**Data collection.** As a starting point for our data collection, we used one year’s worth of production search query traffic<sup>3</sup>. We then performed stratified sampling along the following dimensions: a) search engine, b) activated gender filter on the website, c) query frequency, and d) query length in tokens.

Table 1: Dataset statistics.

Language	Unique pairs	Unique queries	Avg. tokens per query	Unique products
German	10,000	500	3.68	8,076
English	10,000	500	3.99	8,652

<sup>3</sup>Our data collection process complies with the regulations defined in the GDPR and other existing regulatory frameworks around data privacy and safety in the European Union.

Table 2: Agreements between (M)LLM and the human annotator groups (i.e., A1 and A2). We compare agreements based on i) matching either A1 *or* A2 and ii) inter annotator agreement between human annotators (A1 vs. A2) and between LLMs and the human majority vote. In the A1 *or* A2 column, we use the same human majority vote to measure the agreements for human annotators. Results are reported separately for English and German. For human annotations, we report the total time and cost. We use GPT-4o in all steps of our LLM annotation pipeline (Fig. 2). Refer to Table 3 for a more detailed comparison between human annotator groups (A1, A2, and tiebreaker) and different versions of our LLM-powered framework.

Annotator	Agreement with A1 OR A2 (%)		Inter Annotator Agreement (%)		Total time		Cost	
	EN	DE	EN	DE	EN	DE	EN	DE
<b>A1</b>	76.6	75.9	60.2	60.5	3 weeks		€15,000	
<b>A2</b>	75.8	76.1	60.2	60.5	3 weeks		€15,000	
<b>LLM-text</b>	75.4	75.6	62.8	61.4	8 min	7 min	€70	€70
<b>MLLM-text</b>	76.6	77.0	64.3	63.1	12 min	13 min	€92	€93
<b>MLLM-multi</b>	<b>78.1</b>	<b>78.4</b>	<b>65.6</b>	<b>64.7</b>	10 min	10 min	€72	€72
<b>MLLM-multi (gpt-4-turbo)</b>	75.4	75.6	62.9	62.0	30 min	23 min	€155	€156
<i>Without query-specific annotation guidance.</i>								
<b>LLM-text</b>	71.2 <b>(-4.2)</b>	67.0 <b>(-8.6)</b>	57.6 <b>(-5.2)</b>	52.2 <b>(-9.2)</b>	7 min	7 min	€45	€45
<b>MLLM-text</b>	72.5 <b>(-4.1)</b>	68.0 <b>(-9.0)</b>	58.7 <b>(-5.6)</b>	53.4 <b>(-9.7)</b>	11 min	11 min	€66	€67
<b>MLLM-multi</b>	74.1 <b>(-4.0)</b>	71.9 <b>(-6.5)</b>	61.6 <b>(-4.0)</b>	59.9 <b>(-4.8)</b>	4 min	4 min	€25	€24

After sampling and manual curation<sup>4</sup>, we obtained 500 queries in German and English each. For each query, we then used the existing search engine to retrieve 20 products, selecting 15 products from the top of the retrieved results and randomly sampling the remaining 5 from rank 500 onwards. This resulted in 10,000 unique query-product pairs for German and English each. Table 1 outlines the statistics of the collected dataset.

**Data annotation.** For data annotation, we contracted an external agency to facilitate crowdsourcing annotations based on the guidelines we provided.<sup>5</sup> Our requirements specified that only native speakers (German and English, see Table 1) annotate the data. Each query-product pair was to be annotated by two human annotators, with an additional third annotation as a tiebreaker in cases where the two original annotators disagreed. The data annotation process was done in three phases: (i) a pilot phase to onboard annotators and resolve outstanding loopholes and questions in the annotation guidelines; (ii) the main annotation phase; (iii) the tiebreaker phase.

In total, the data annotation process – from the handover of the initial version of the annotation guidelines by us to the final delivery of annotated data by the external agency – took approximately 8

weeks<sup>6</sup>, with roughly 3 weeks needed for completing the main annotation and tiebreaker phases. The total cost of data annotation was €30,000.

### 3.2 LLM versus Human Annotators

Using our proposed framework, we assessed the relevance quality of the 20,000 unique query-product pairs. Table 2 summarizes the results of our experiments using few-shot prompting, where we incorporated examples into the system prompts of the (M)LLMs. Here, we provided the LLMs with example customer queries, their corresponding requirement lists, and quality labels, but not complete product descriptions or images.

Initially, we randomly sampled 100 examples from the English dataset and examined the relevance labels assigned by both LLMs and human annotators. We used the results of this step to adjust the few-shot examples in the system prompt.

We compare the performance of different versions of our pipeline with human annotations. In Table 2, these versions are labelled as “LLM-text”, “MLLM-text”, and “MLLM-multi”.

“LLM-text” is the simplest version where only product descriptions in textual form are used, without incorporating product images. In “MLLM-text”, we employ a vision model to generate textual descriptions of product images (Step 4 in the pipeline, see Fig. 2). The generated textual description of the product image is then concatenated with the

<sup>4</sup>We manually checked all sampled queries to ensure we cover a diverse and broad spectrum. For example, we would manually replace *yet another* sneakers-related query with a boots-related query of the same length.

<sup>5</sup>See Appendix B for an overview of our annotation guidelines.

<sup>6</sup>This excludes the effort needed to write the annotation guidelines and to sample the query-product candidate pairs.

product description itself (Step 5a). “MLLM-multi” utilises the same textual input as “LLM-text”, while also incorporating the product image as an additional input.<sup>7</sup> Comparing “MLLM-multi” and “LLM-text” highlights the impact of multimodal inputs on our task.

In all cases, the (M)LLM uses product information (in different modalities, depending on the version), query requirements, and query-specific annotation guidance to assign relevance labels.

Overall, Table 2 shows that the agreement between human annotators and LLMs is on par with that between human annotators, supporting the scalability of LLM annotation for production-level traffic.

Table 2 also shows the results of an ablation study that removed the query-specific annotation guideline (Step 1 in Fig. 2). The inclusion of this guideline improved agreements by approximately 4-10%. More importantly, this component in our framework is essential for enhancing the interpretability and debugging of LLM-based decisions. However, as expected, incorporating query-specific annotation guidelines and chain-of-thought reasoning increased the evaluation costs.

We also tested the impact of different (M)LLM architectures in our pipeline.<sup>8</sup> The results shown in Table 2 are based on “GPT-4o” (OpenAI, 2024) except for one row, labelled “MLLM-multi (gpt-4-turbo)”. In the case of GPT-4 Turbo, the agreement with human annotators consistently fell below that of GPT-4o, while its costs and evaluation times exceeded those of all other architectures.

In Table 4, we repeated the experiments using GPT-3.5 Turbo. As expected, the results were significantly worse compared to GPT-4o or GPT-4 Turbo. However, the cost and time required for GPT-3.5 Turbo were lower than for the other architectures.

## 4 Discussion

**Agreement between LLM and human annotators.** The human-LLM agreements between “MLLM-multi” and the human majority vote – 65.6% for EN and 64.7% for DE in Table 2 – are in line with the human inter-annotator agreement, which is 60.2% for EN and 60.5% for DE.

To better identify discrepancies between LLM and human annotations, we focused our analysis on hard disagreements between the two. We consider

<sup>7</sup>See the orange rectangle in Fig. 2 where both the textual product description and its image are fed into an MLLM.

<sup>8</sup>Refer to Table 3 and 4 in Appendix C for the results of similar experiments conducted with different LLM types.

a hard disagreement to be when, for example, the LLM considers a product to be “highly relevant” for the given query, whereas the human majority judgement would be “irrelevant”, and vice versa. In total, we found that approximately 15% of annotations<sup>9</sup> in our dataset were hard disagreements. For manual analysis, we sampled 20% of the hard disagreements and found that in 50% of the cases, the human annotation was wrong, in 31% the LLM was wrong, and in 19% of cases, both the LLM and the humans provided a wrong annotation.

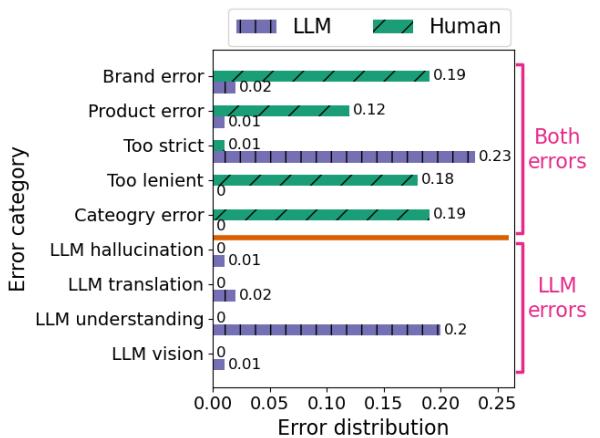


Figure 3: Distribution of errors between LLMs and humans on hard disagreements (50% were due to human errors, 31% LLM errors and in 19% both made an error). The upper part (“Both errors”) focuses on errors that either the LLM or humans could make. It highlights that LLMs and humans make very different types of errors. In addition, the lower part (“LLM errors”) shows the distribution of errors that only an LLM would make. Predominantly these are misunderstandings of a part of the search query.

We further categorised the hard disagreements into 9 error classes<sup>10</sup>, and found LLMs and humans tend to make very different kinds of errors. For example, as shown in Fig. 3, the main errors made by the LLM are either being *too strict* in their judgement (e.g., considering a product as “irrelevant”, where “acceptable substitute” would have been more appropriate), or misunderstanding a part of the query (e.g., interpreting *On Vacation* in its literal sense rather than the fashion brand). On the other hand, humans would oftentimes be *too lenient* when LLMs were *too strict* (e.g., considering a product as “highly relevant” when “accept-

<sup>9</sup>2,971 out of 20,000 query-product pairs ( $\approx 15\%$ ) had hard disagreements between the LLM and human majority vote.

<sup>10</sup>See Appendix D for the definition of these classes and for example errors we have observed.

able substitute” would have been more appropriate). Furthermore, human annotations frequently exhibited brand errors (e.g., considering a pair of *Lee* jeans as “highly relevant” for a query requesting *Levi’s jeans*), product errors (e.g., considering an *Adidas Samba* sneaker as “highly relevant” for a query requesting an *Adidas Stan Smith* sneaker), or category errors (e.g., considering a pair of *Nike shirts* as “highly relevant” for a query requesting *Nike shoes*), which we barely ever observed for LLMs. We hypothesise that the latter three kinds of human errors are primarily due to annotation fatigue as specifically these cases have been prominently and unambiguously featured in the annotation guidelines.

These findings suggest that LLMs might be a more reliable source for the bulk of annotations, freeing human labour to focus on trickier cases.<sup>11</sup> In the human-machine collaboration spectrum introduced by Faggioli et al. (2023), our approach can be classified as a “Human Verification” (or human-in-the-loop) approach.

#### Subjective nature of relevance judgements.

We found that human disagreement was dominated by two main factors, (i) human errors due to annotation fatigue as described above<sup>12</sup>, and (ii) the inherent subjective nature of the task. For the latter, we attribute the source of disagreement to either the ambiguity in the annotation guidelines (even comprehensive guidelines cannot cover all possible cases), or to the subjective judgement of the annotator.<sup>13</sup> Ideally, the annotation guidelines should make the task as objective as possible; however, in practice, there is always a level of subjectivity.

**Annotation time and cost.** (M)LLMs are approximately 100 to 1,000 times cheaper than human annotators, and the time required to complete all 20,000 annotations using (M)LLMs is significantly smaller (around 20 minutes for (M)LLMs compared to about 3 weeks for human annotators). Note that several human annotators worked in each group (i.e., “A1” and “A2” in Table 2), and the total time reported in Table 2 is for annotating all query-product pairs. This excludes the time spent

<sup>11</sup>This typically includes new trending terms or styles that LLMs have not observed yet, but also ambiguous queries such as *old money* or *dark academia*, that specify a fashion style.

<sup>12</sup>In fact we found that human-human hard disagreements, also making up approx. 15% of the annotated data, were also primarily due to brand errors, product errors and category errors — the same types of errors that we also found when comparing LLM and human hard disagreements.

<sup>13</sup>See Appendix E for examples.

on scoping and onboarding human annotators. For (M)LLMs, the reported time excludes the pipeline development time and only includes the actual annotation time.

We anticipate that both cost and time will decrease even further as LLMs and their APIs become more efficient. Moreover, new approaches, such as batch processing, can further reduce costs (e.g., OpenAI’s new batch processing is half the price of non-batch queries<sup>14</sup>). Indeed, in production, we use batch processing to assess query-product pairs across markets on a nightly basis.

**Relevance assessment in production.** High relevance is a necessary, but not a sufficient condition, for high customer engagement, as it is also determined by other factors, e.g. personal preferences, product availability, and price expectations. In this paper, we focus on semantic relevance, but in production we rank the retrieved documents based on various features to take into account both relevance to the query and customers’ personal preferences.

Currently, we use the LLM-powered evaluation framework presented in this paper in production to continuously perform relevance assessments at scale. We typically focus on monitoring the performance of high-volume queries with our framework. Additionally, we evaluate the retrieval performance for low-performing queries. We identify such queries based on signals indicating low relevance in top ranked results, such as low engagement with the result set and high friction in customer experience (e.g., a high reformulation rate<sup>15</sup>) or high exit rate. This approach enables us to significantly reduce costs and to enhance customer experience faster by prioritising the queries that need the most attention and optimising our resources accordingly.

## 5 Conclusion

Our novel evaluation method leveraging Multi-modal LLMs demonstrates a highly efficient approach to assessing large-scale IR systems in product retrieval. We introduce query-level annotation guidelines for calibration and utilise the multi-modal capabilities of foundation models to assess the relevance of retrieved products for a query. Our

<sup>14</sup><https://platform.openai.com/docs/guides/batch> (accessed on 2024-07-17)

<sup>15</sup>The reformulation rate is the percentage of queries that are modified and resubmitted within the same search session, indicating an initial failure to satisfy the user’s intent and a subsequent attempt to refine the search.

LLM-powered framework, combined with caching and parallel processing, leads to significant reductions in both time and cost. The method’s scalability, ability to handle multilingual queries and products, and support for continuous offline evaluations are crucial for large IR systems operating in diverse markets. Experimental results, validated against 20,000 human annotations, confirm the effectiveness and efficiency of our approach. A detailed analysis of human and (M)LLM annotations indicates that (M)LLMs are a more reliable source for relevance assessment in large-scale IR systems. We are currently leveraging this framework in production to continuously perform relevance assessments at scale and maintain a high-quality user experience. Additionally, we utilise its outputs to train, evaluate, and analyse other components of our search and ranking systems.

## 6 Ethics Statement

Our data collection process strictly adheres to the General Data Protection Regulation (GDPR) and other relevant data privacy and safety laws within the European Union. We ensure that all data utilised, including human evaluation data, is anonymised to safeguard against the disclosure of any personally identifiable information.

We do not suggest replacing human annotators with large language models (LLMs). Instead, we focus on leveraging the strengths of both. Our analysis indicates that human annotators may make errors due to annotation fatigue or lack of domain knowledge—errors not observed with LLMs. Therefore, we recommend using LLMs for bulk annotation work while reserving human expertise for more complex cases.

We are committed to advancing responsible and unbiased AI technologies and welcome any inquiries regarding the ethical aspects of our work.

## References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Omar Alonso and Stefano Mizzaro. 2012. Using crowdsourcing for trec relevance assessment. *Information processing & management*, 48(6):1053–1066.

Jo Kristian Bergum. 2024. [Improving retrieval with llm-as-a-judge](#). Accessed: 2024-07-10.

Roi Blanco, Harry Halpin, Daniel M Herzig, Peter Mika, Jeffrey Pound, Henry S Thompson, and Thanh Tran Duc. 2011. Repeatable and reliable search system evaluation using crowdsourcing. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 923–932.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Kuan-Ta Chen, Omar Alonso, Martha Larson, and Irwin King. 2016. Introduction to the special issue on crowd in intelligent systems.

Yan Chen, Shujian Liu, Zheng Liu, Weiyi Sun, Linas Baltrunas, and Benjamin Schroeder. 2022. Wands: Dataset for product search relevance assessment. In *European Conference on Information Retrieval*, pages 128–141. Springer.

Guglielmo Faggioli, Laura Dietz, Charles L. A. Clarke, Gianluca Demartini, Matthias Hagen, Claudia Hauff, Noriko Kando, Evangelos Kanoulas, Martin Potthast, Benno Stein, and Henning Wachsmuth. 2023. Perspectives on large language models for relevance judgment. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR ’23*, page 39–50, New York, NY, USA. Association for Computing Machinery.

Marco Ferrante, Nicola Ferro, and Maria Maistro. 2017. Aware: exploiting evaluation measures to combine multiple assessors. *ACM Transactions on Information Systems (TOIS)*, 36(2):1–38.

Martin Halvey, Robert Villa, and Paul D. Clough. 2015. [Sigir 2014: Workshop on gathering efficient assessments of relevance \(gear\)](#). *SIGIR Forum*, 49(1):16–19.

Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M Henne. 2009. Controlled experiments on the web: survey and practical guide. *Data mining and knowledge discovery*, 18:140–181.

Matthew Lease and Emine Yilmaz. 2013. Crowdsourcing for information retrieval: introduction to the special issue. *Information retrieval*, 16:91–100.

Adam Marcus, Aditya Parameswaran, et al. 2015. Crowdsourced data management: Industry and academic perspectives. *Foundations and Trends® in Databases*, 6(1-2):1–161.

Wendy W. Moe. 2003. [Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream](#). *Journal of Consumer Psychology*, 13(1-2):29–39.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2021. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*.

OpenAI. 2024. [Hello gpt-4o](#). Accessed: 2024-07-17.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Hossein A. Rahmani, Nick Craswell, Emine Yilmaz, Bhaskar Mitra, and Daniel Campos. 2024. [Synthetic test collections for retrieval evaluation](#). *Preprint*, arXiv:2405.07767.

Mark Sanderson et al. 2010. Test collection based evaluation of information retrieval systems. *Foundations and Trends® in Information Retrieval*, 4(4):247–375.

Beatriz Soviero, Daniel Kuhn, Alexandre Salle, and Vianne Pereira Moreira. 2024. Chatgpt goes shopping: Llms can predict relevance in ecommerce search. In *Advances in Information Retrieval*, pages 3–11, Cham. Springer Nature Switzerland.

Karen Spark-Jones. 1975. Report on the need for and provision of an 'ideal' information retrieval test collection. *Computer Laboratory*.

Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2023. Large language models can accurately predict searcher preferences. *arXiv preprint arXiv:2309.10621*.

Shivani Upadhyay, Ehsan Kamaloo, and Jimmy Lin. 2024. Llms can patch up missing relevance judgments in evaluation. *arXiv preprint arXiv:2405.04727*.

Ellen M Voorhees. 2001. The philosophy of information retrieval evaluation. In *Workshop of the cross-language evaluation forum for european languages*, pages 355–370. Springer.

Haixun Wang and Taesik Na. 2024. [Rethinking e-commerce search](#). *SIGIR Forum*, 57(2).

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*, 35:24824–24837.

## A Multimodal LLM-powered relevance assessment: evaluation steps for an example query

Fig. 4 illustrates the various steps of our evaluation framework using the example query *women’s long sleeve t-shirt with green stripes*.

Given this query, the LLM infers its requirements and their importance (Step 1 of our framework, refer to Fig. 2). The outputs of this step for the example query are detailed in the paragraph “Query requirements and their importance” in Fig. 4b. As shown in Fig. 4b, the LLM has inferred four query requirements: “assortment category”, “sleeve length”, “product type” and “pattern”. An importance level is also assigned to each requirement (in this case, the first three requirements are “must\_have”, and the last one is “approximate\_is\_okay”). Additionally, the LLM provides a reason for each requirement and its importance (not shown here). The LLM also translates the query into English and assigns a “specificity” level, as shown in Fig. 4b.

In panel (c), the translated query, its specificity, its requirements and their importance are used to create *query-specific annotation guidelines*. The three quality labels (i.e., “irrelevant”, “acceptable\_substitute” and “highly\_relevant”) are pre-defined. However, the guidance for each label is generated by the LLM. The LLM provides clear and detailed descriptions for each relevance label, tailored to the given query. In the ablation study of Table 2, we assessed the impact of query-specific annotation guidelines on our method’s performance. To do this, we replaced the query-specific guidelines with a generic one, as shown in Fig. 5.

In Fig. 4d, an example product, its attributes, and its image are shown. These attributes are read from an existing database and are not generated by the LLM, except for the “visual description of packshot”, highlighted by a red rectangle which is generated by a vision model (e.g., GPT-4o). The (M)LLM uses the query-specific annotation guidance in panel (c), along with the extracted and generated product attributes in panel (d), to assign a relevance label. In this example, as shown in panel (e), the label is “highly\_relevant”, and the reasoning (aka the chain-of-thought step) of the (M)LLM is shown for inspection and debugging purposes.

## B Human Annotation Guidelines

For human annotators, we focused on three classes:

- **highly relevant:** The retrieved product satisfies all the specifications in the query.
- **acceptable substitute:** The item fulfils some, but not all aspects of the query and the retrieved item can be used as a functional substitute.
- **irrelevant:** A central aspect of the query is not fulfilled (e.g. wrong brand, wrong category, wrong product).

We decided against a more granular annotation scale to reduce mental load on annotators and to (hopefully) harness higher agreement scores among annotators.

Our annotation guidelines also reflect requirements that are more business-specific rather than content-specific. For example, annotators have been explicitly briefed that if a query requests a specific brand (e.g. *Polo Ralph Lauren jumpers*), any retrieved item that is not from the requested brand is to be regarded as “irrelevant”.

Another business-specific rule was that if a query requests a particular product (e.g. *The North Face 1996 retro nuptse jacket*), any retrieved product that is not that particular type of The North Face jacket is to be regarded as “irrelevant”.

Despite the explicit mentions of these rules, numerous provided examples across product categories, and an additional briefing session after the annotation pilot phase, brand and product errors were among the most commonly made human annotation errors.

## C Experiments with LLM types: GPT-3.5, GPT-4, and GPT-4o

In this section, we compare human annotator groups with (M)LLMs using different architectures. The results in Table 3 are primarily based on “GPT-4o” (OpenAI, 2024), except for the row labelled “MLLM-multi (gpt-4-turbo)”. For GPT-4 Turbo, the agreement with human annotators was consistently lower than that of GPT-4o, while its costs and evaluation times exceeded those of all other architectures.

In Table 4, we repeated the experiments using GPT-3.5 Turbo. As expected, its results were significantly worse compared to GPT-4o and GPT-

4 Turbo, but its cost and time requirements were lower than those of the other architectures.

## D LLM versus Human error types

After manually inspecting a sample of hard disagreements<sup>16</sup>, we defined the following 9 error classes, some of which are applicable to LLMs and humans, and some to LLMs only:

**1. Brand error.** When a user specifies a brand name in the search query, e.g. *Lee jeans*, *Nike sneakers*, or *Mascara dresses*, we consider any retrieved item as “irrelevant” if it is not from the requested brand. This is independent of whether the retrieved item would be visually similar to the requested one. This requirement has been covered in the LLM prompt as well as the human annotation guidelines. Predominantly, this error has been made by human annotators (see Fig. 3).

**2. Product error.** When a user specifies a specific product in the search query, e.g. *Levis 501* or *Adidas Stan Smith*, we consider any retrieved item that is not exactly the requested item as “irrelevant”. This requirement has been covered in the LLM prompt as well as the human annotation guidelines. Predominantly, this error has been made by human annotators (see Fig. 3).

**3. Too strict.** This error happened when a product was judged as “irrelevant” for a given query despite fulfilling *almost* all the requirements of the query. This error has been predominantly made by LLMs (see Fig. 3), for example when a query requested *black Levis jeans with holes*, but the retrieved product was a grey pair of Levis jeans with holes, the LLM would typically annotate the retrieved products as “irrelevant”.

**4. Too lenient.** This error happened when a product was judged as “highly relevant” for a given query, despite not fulfilling all requirements that the query specified. This error has been exclusively made by human annotators (see Fig. 3), for example where for a query like *Nike Air Force One high-top*, humans annotated a *Nike Air Force One low-top* sneaker as “highly relevant”.

**5. Category error.** When a user specifies the category of a fashion item in the search query, e.g. *dress*, *sneakers*, *belts*, we consider any retrieved item that does not match the category as “irrelevant”. This requirement has been covered

<sup>16</sup>We consider a hard disagreement to be when, for example, the LLM considers a product to be “highly relevant” for a given query, whereas the human majority judgement would be “irrelevant”, and vice versa. Refer to Section 4 for details.

in the LLM prompt as well as the human annotation guidelines. Predominantly, this error has been made by human annotators (see Fig. 3).

**6. LLM hallucination error.** We rarely observed hallucinations as a source of error. Interestingly, when hallucinations did occur, they were exclusively related to size queries, such as *t-shirt xxl*. In such cases, the LLM would hallucinate various available sizes for a given retrieved product and make a relevancy judgement on the basis of its hallucinations.

**7. LLM translation error.** Since our dataset contained German and English queries, the LLM was prompted to translate a German query into English before starting its reasoning process. This sometimes resulted in translation errors that subsequently led to incorrect relevancy judgements. For example, it happened for queries containing the term *Unterziehhoose*, meaning some sports leggings one can wear underneath sports shorts, which the LLM incorrectly translated as *underpants*.

**8. LLM understanding error.** This error category is somewhat broader. We would categorise an LLM error as *understanding error*, whenever the LLM misinterpreted a part of the query or the product. For example, this error occurred when the LLM would misinterpret a query for *Nike Tech Fleece* to be focused on the material whereas *Tech Fleece* typically refers to a particular series of Nike sports clothing. Another example is the misinterpretation of brand names, such as for *On Vacation* (interpreted in its literal meaning), or for *Evry Jewels* (where *Evry* would be interpreted to mean *Every*). To our amusement during error analysis, we also observed a brand misinterpretation for the query *miniature winter jackets for kids*, where *Mini A Ture* is a kids’ clothing brand. The LLM interpreted *miniature* in its literal sense and reasoning that [...] the sizes available are for kids, which fits the ‘miniature’ requirement.

**9. LLM vision error.** Some of our models included the visual interpretation of a product image in its relevancy assessment.<sup>17</sup> We only rarely observed LLM vision errors. If they did occur, it was typically when the product image was taken at a slight angle—for instance, with a pair of sneakers where the LLM erroneously identified them as high-top due to the photo angle. Errors were also more likely when the image included a human model, which acted as a distractor.

<sup>17</sup>MLLM-text in Tables 2, 3 and 4.

## E Subjective Nature of Relevance Judgements

The difficulty in judging query-product relevancy can vary widely. For example, for queries such as *Nike Air Max 95* or *Paul Smith long sleeve polo shirt*, there is barely any room for subjectivity — the retrieved products either are matches, or they are not. And indeed, this is reflected in the human-human inter-annotator agreement (95% and 82%, respectively) and the LLM-human agreement (98% and 89%, respectively), for these two examples.

However, there are numerous queries that are much more open to subjective judgement. One such example is the query *smart casual shoes*, where the human-human agreement was only 12% and the LLM-human agreement was 24%. The range of suitable products for this query spans various types of shoes, and whether or not a particular shoe can be categorised as *smart casual* is typically not included in the product data. In these cases, humans and LLMs would draw on their prior knowledge for making a relevance judgement. LLMs would generally be a stricter judge and consider anything that resembles a sneaker too closely, or is not in a shade of black or brown, as “irrelevant”. Human strictness for relevancy judgements for this query varied between the very formal and the (loosely speaking) anything goes extremes.

You are an expert query-SKU relevance evaluator. You will be provided with a sku description. Your task is to evaluate how closely the SKU matches the search query.

| **Search query:** women's long sleeve t-shirt with green stripes

(a)

**Query translated:** women's long sleeve t-shirt with green stripes

**Query specificity:** specific

**Query requirements and their importance:**

- \* assortment category: women's | importance: must\_have
- \* sleeve length: long sleeve | importance: must\_have
- \* product type: t-shirt | importance: must\_have
- \* pattern: green stripes | importance: approximate\_is\_okay

(b)

**Annotation guidelines:**

You can select one of the following relevance quality labels:

- \* **irrelevant:** Items that are not women's clothing, not t-shirts, or do not have long sleeves. For example, men's t-shirts, women's short sleeve t-shirts, or women's t-shirts without any stripes.
- \* **acceptable\_substitute:** Women's long sleeve t-shirts with a similar pattern, such as different shades of green stripes or a slightly different stripe pattern.
- \* **highly\_relevant:** Women's long sleeve t-shirts with green stripes that match the query exactly.

(c)

**Product information:**

[Article SKU]



(d)

\* **Description (from product page):**

name: [Brand Name] | color: evergreen | gender group: Women | age group: Adult | silhouette code: t\_shirt\_top | product group: clothing | category tag: Long sleeved top, Langarmshirt, Long sleeved top | fashion tags: shirt, knits, top, long sleeves, clothing | material: rib | upper material: elastane, cotton | shape: straight | fit taxonomy: Regular Fit | pattern: striped | size: L, M, S, XL, XS, XXL | occasion: leisure | neckline: crew neck | clothing length: normal | sleeve length option: long | fabric definition: knitted |

**Visual description of packshot:** This long-sleeved top features a classic horizontal stripe pattern in alternating white and green. The fabric has a ribbed texture, giving it a subtle, snug fit that accentuates the wearer's shape. The neckline is a simple, rounded crew neck, contributing to a timeless and versatile look. Its overall design is casual yet stylish, perfect for everyday wear.

Vision model

**Relevance assessment:**

**Relevance label:** 'highly\_relevant'

(e)

**Reasoning (chain-of-thought):** The SKU description provided is for a women's long-sleeved top, which matches the 'women's' and 'long sleeve' requirements. The product type is described as a 't-shirt top' and 'long sleeved top,' which aligns with the 't-shirt' requirement. The pattern is described as 'striped' with alternating white and green, which matches the 'green stripes' requirement. Therefore, all must-have attributes and the approximate attribute are met.

Figure 4: Evaluation steps for an example query *women's long sleeve t-shirt with green stripes*. The entire content displayed in this figure is generated by Multimodal LLMs, except for panel (a), the packshot in panel (d), and the black dashed rectangle also in panel (d). However, within the attributes shown in panel (d), the "visual description of packshot", highlighted by a red rectangle, is also generated by a vision model (specifically, GPT-4o was used in this instance). Please refer to the text for further details. (In this example, we have removed the brand name from the product description and the tag on the packshot.).

<b>Annotation guidelines:</b>												
You can select one of the following relevance quality labels:												
* <b>irrelevant</b> : the item is completely irrelevant to the query.												
* <b>acceptable_substitute</b> : Not all query requirements are fulfilled but the customer could still be okay with seeing this result.												
* <b>highly_relevant</b> : the item is completely relevant to the query.												

Figure 5: A *generic* annotation guideline for the task of query-product relevance assessment. Compare this to the *query-specific* annotation guidelines in panel (c) of Fig. 4.

Table 3: Agreements between (M)LLM and the human annotator groups (i.e., A1, A2 and tiebreaker). Similar to Table 2, but with additional columns showing the agreements of A1, A2, and the tiebreaker groups with other annotators. We report agreements separately for English and German. For human annotations, we report the total time and cost. We use GPT-4o in all steps of our LLM annotation pipeline (Fig. 2).

Annotator	Agreement with A1 OR A2 (%)		Agreement with A1 (%)		Agreement with A2 (%)		Agreement with Tiebreaker (%)		Inter Annotator Agreement (%)		Total time		Cost	
	EN	DE	EN	DE	EN	DE	EN	DE	EN	DE	EN	DE	EN	DE
<b>A1</b>	76.6	75.9	-	-	<b>60.2</b>	60.5	40.0	37.5	60.2	60.5	3 weeks		€15,000	
<b>A2</b>	75.8	76.1	60.2	60.5	-	-	37.5	36.6	60.2	60.5	3 weeks		€15,000	
<b>LLM-text</b>	75.4	75.6	63.9	60.8	55.8	59.3	41.9	33.9	62.8	61.4	8 min	7 min	€70	€70
<b>MLLM-text</b>	76.6	77.0	64.9	62.0	56.9	60.7	43.2	35.5	64.3	63.1	12 min	13 min	€92	€93
<b>MLLM-multi</b>	<b>78.1</b>	<b>78.4</b>	<b>66.0</b>	<b>63.1</b>	58.2	<b>62.0</b>	<b>43.9</b>	37.4	<b>65.6</b>	<b>64.7</b>	10 min	10 min	€72	€72
<b>MLLM-multi (gpt-4-turbo)</b>	75.4	75.6	63.9	61.1	55.8	59.3	41.7	34.8	62.9	62.0	30 min	23 min	€155	€156
<i>Without query-specific annotation guidance</i>														
<b>LLM-text</b>	71.2	67.0	57.5	52.9	54.4	51.4	39.0	34.4	57.6	52.2	7 min	7 min	€45	€45
<b>MLLM-text</b>	72.5	68.0	58.8	53.8	55.4	52.4	39.7	34.1	58.7	53.4	11 min	11 min	€66	€67
<b>MLLM-multi</b>	74.1	71.9	61.1	57.1	56.3	56.4	41.9	<b>42.3</b>	61.6	59.9	4 min	4 min	€25	€24

Table 4: Like Table 3, except we use GPT-3.5 Turbo (specifically, “gpt-3.5-turbo-0125”) for text inputs and GPT-4o for generating textual descriptions for image inputs (Step 4 in Fig. 2). Here, we do not have “MLLM-multi” as GPT-3.5 Turbo does not accept multimodal (text and image) inputs.

Annotator	Agreement with A1 OR A2 (%)		Agreement with A1 (%)		Agreement with A2 (%)		Agreement with Tiebreaker (%)		Inter Annotator Agreement (%)		Total time		Cost	
	EN	DE	EN	DE	EN	DE	EN	DE	EN	DE	EN	DE	EN	DE
<b>A1</b>	<b>76.6</b>	75.9	-	-	<b>60.2</b>	<b>60.5</b>	40.0	37.5	<b>60.2</b>	<b>60.5</b>	3 weeks		€15,000	
<b>A2</b>	75.8	<b>76.1</b>	<b>60.2</b>	<b>60.5</b>	-	-	37.5	36.6	<b>60.2</b>	<b>60.5</b>	3 weeks		€15,000	
<b>LLM-text</b>	65.2	63.4	51.3	48.5	49.6	48.7	40.1	37.7	52.1	49.3	6 min	6 min	€6	€6
<b>MLLM-text</b>	66.1	62.6	51.6	48.2	50.9	48.1	40.8	39.2	53.1	49.6	11 min	11 min	€24	€25
<i>Without query-specific annotation guidance</i>														
<b>LLM-text</b>	59.1	57.0	43.9	43.0	46.5	42.8	41.1	47.9	47.6	46.3	4 min	4 min	€4	€4
<b>MLLM-text</b>	60.1	57.2	45.0	42.9	47.4	43.4	<b>41.9</b>	<b>50.2</b>	48.9	47.4	9 min	10 min	€21	€22