INSTRUCTIR: A Benchmark for Instruction Following of Information Retrieval Models

Hanseok Oh[1]* **Hyunji Lee**[2] **Seonghyeon Ye**[2] **Haebin Shin**[2]
Hansol Jang[3] **Changwook Jun**[3] **Minjoon Seo**[2]
[1]SoftlyAI [2]KAIST AI [3]LG AI Research
hanseok.oh@softly.ai

Abstract

Despite the critical need to align search targets with users' intentions, retrievers often only prioritize query information without delving into the users' intended search context. Enhancing the capability of retrievers to understand the intentions and preferences of users, akin to language model instructions, has the potential to yield more aligned search targets. Prior studies restrict the application of instructions in information retrieval to a task description format, neglecting the broader context of diverse and evolving search scenarios. Furthermore, the prevailing benchmarks utilized for evaluation lack explicit tailoring to assess instruction-following ability, thereby hindering progress in this field. In response to these limitations, we propose a novel benchmark, INSTRUCTIR, specifically designed to evaluate instruction-following ability in information retrieval tasks. Our approach focuses on user-aligned instructions tailored to each query instance, reflecting the diverse characteristics inherent in real-world search scenarios. Through experimental analysis, we observe that some retrievers fine-tuned to follow task-style instructions, such as INSTRUCTOR (Su et al., 2022), can underperform compared to their non-instruction-tuned counterparts. This underscores potential overfitting issues inherent in constructing retrievers trained on existing instruction-aware retrieval datasets [1].

1 Introduction

Large Language Models (LLMs) are often further trained to align user instructions and preferences with instruction tuning for diverse generative tasks (Ouyang et al., 2022; Wang et al., 2022a; Zhang et al., 2023b). This kind of alignment to user preferences is also important for information retrievers to reflect diverse users' search intentions and preferences for the search targets. For example, when a user writes a blog post for children about the current climate change issue, it may be better to search for articles that are easy to understand rather than complex scientific articles. However, current retrievers often do not take this into account, focusing on utilizing only ambiguous queries even simplifying the details for users through reformulation (Ma et al., 2023a). Moreover, lack of benchmarks to evaluate retrievers on user-aligned scenarios prevents the mature discussions of instruction following in retrieval task.

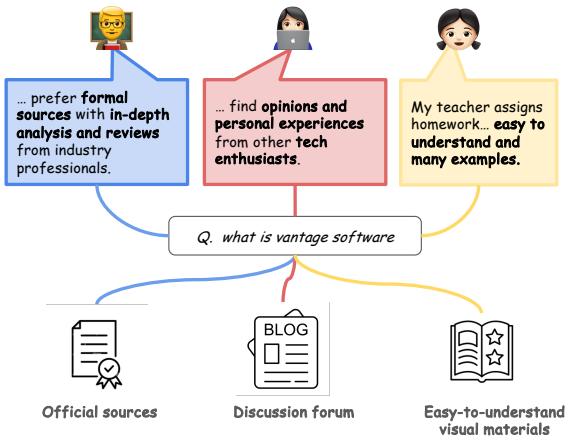

Figure 1: INSTRUCTIR benchmark is designed to evaluate instruction following ability in information retrieval tasks. As unique user-aligned instructions change, different search targets should be retrieved to reflect real-world search scenarios.

In order to effectively reflect the various intentions and situations that real-world users actually ask, employing instance-wise instructions for queries is more appropriate than relying on coarse-grained instructions that share the same task-specific guidance for various queries (Wang et al., 2022a; Ye et al., 2023). Several studies explore the integration of instructions into retrievers, but they primarily concentrate on building general purpose retrievers, which often limit the examination

* Most work performed at KAIST AI and during internship at LG AI.

[1]Code and dataset are available at https://github.com/kaistAI/InstructIR

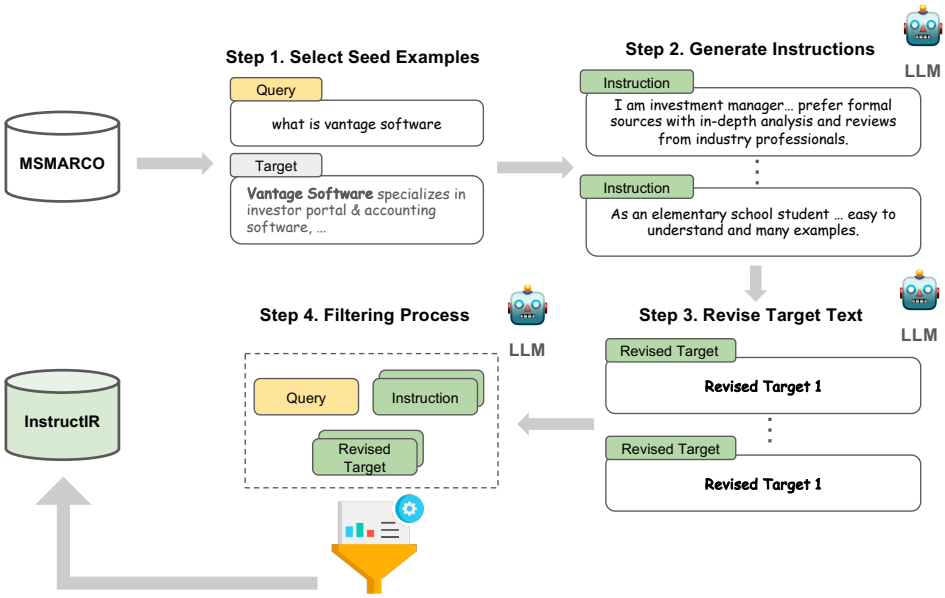

Figure 2: Overview of data creation pipeline for building INSTRUCTIR benchmark. To build datasets that demand diverse user-aligned instructions for each query, we begin by selecting seed examples from the MSMARCO datasets. Subsequently, we generate a variety of instructions suitable for each query, revise the target text to align with these instructions, and systematically filter the generated content. The resulting dataset is used for INSTRUCTIR benchmark. GPT-4 is employed in this generation pipeline.

to task-specific instructions (Asai et al., 2023; Su et al., 2022; Wei et al., 2023). For instance, these studies uniformly apply the same instructions to all instances, presenting them in the form of task descriptions, such as *"Search for the Wikipedia paragraph that answers this question"*. Furthermore, the evaluation of these instruction-tuned models relies on benchmarks that inherently do not mandate instructions for task resolution (Thakur et al., 2021; Santhanam et al., 2021; Muennighoff et al., 2022). Due to these limitations, the extent to which retrievers can effectively follow instructions has not been thoroughly evaluated.

In this work, we introduce a novel benchmark, INSTRUCTIR, specifically designed to evaluate instruction-following ability of retrieval models with diverse user-aligned instructions for each query, mirroring real-world search scenarios. We collect a total of 9,906 of *instance-wise instructions* that involve details about the search users, such as their job, background, situation, location, hobbies, additional interests, and search goals, and preferred sources. Notably, INSTRUCTIR stands out from other benchmarks that evaluate task-aware instructions due to the distinctiveness in instruction types and diversity, as delineated in Table 1. Instructions and corresponding search targets are acquired through our multi-stage data creation pipeline and filtering process, leveraging GPT-4 (OpenAI, 2023), as illustrated in Figure 2. The quality of the datasets is verified through a combination of human evaluation and machine filtering, resulting in a high-quality dataset. Additionally, we introduce the Robustness score as an evaluation metric, quantifying the ability of retrievers to robustly follow instructions. These metrics offer a holistic perspective on how effectively retrievers adapt to changes while conveying the same query with varying instructions.

We evaluate over 12 retriever baselines on INSTRUCTIR including both naïve retrievers (not explicitly instruction-tuned) and instruction-tuned retrievers. With our experiments, we find that task-style instruction-tuned retrievers, such as INSTRUCTOR (Su et al., 2022), consistently underperform compared to their non-tuned counterparts, which cannot be found with the previous benchmarks. Notably, utilizing an instruction-tuned language model and larger model as the backbone demonstrates the most potent performance improvement. Through INSTRUCTIR, we gain valuable insights into the diverse characteristics of existing retrieval systems. We anticipate that this benchmark will contribute to accelerating progress in the development of more sophisticated, controllable, and instruction-aware information access systems.

2 Related Works

Evaluation for Instruction Following. Insturction tuning is a crucial technique to enhance the capabilities and controllability of large language models (LLMs). Instruction tuning refers to the process of further training LLMs on a dataset consisting of *(instruction, output)* pairs in a supervised fashion, which bridges the gap between the next-word prediction objective of LLMs and the users' objective of having LLMs adhere to human instructions (Ouyang et al., 2022; Zhang et al., 2023a). For the approaches to evaluate instruction following capabilities in generative tasks can be categorized as follows: Instructions for cross-task generalization, User Aligned instructions, and Verifiable instructions. **Instructions for cross-task generalization** focus on evaluating cross-task generalization under instructions training models to follow instructions on a subset of tasks and evaluating them on the remaining unseen ones (Wang et al., 2022b; Liang et al., 2022; Wang et al., 2022b). **User Aligned instructions** focus on evaluating how instruction-based models handle diverse and unfamiliar instructions (Ye et al., 2023; Wang et al., 2022a; Dubois et al., 2023). Unlike coarse-grained evaluation as for cross-task generalization instructions, they are different per instance. **Verifiable instructions** are focusing on straightforward and easy-to reproduce evaluation benchmark that focuses on a set of "verifiable instructions" (Efrat et al., 2022; Zhou et al., 2023; Jiang et al., 2023).

Instruction Following in Information Retrieval. Building instruction-tuned models on text embedding tasks is mostly focused on building general purpose model that can solve multiple tasks with task description as an instructions (Asai et al., 2023; Su et al., 2022; Wei et al., 2023). Models are trained with multiple source of train dataset with task descriptions as instructions, such as BERRI and MEDI, and evaluated on the held out tasks which haven't seen during the train time. However, for solving these tasks it is not essential to follow instructions. The relationship between query and targets are already in one to one relationship, where hard to evaluate effect of instructions. Moreover, the benchmarks used for evaluating instruction following ability are BEIR (Thakur et al., 2021), LoTTE (Santhanam et al., 2021), X2 (Asai et al., 2023), which are not suitable for evaluating the retrieval model's ability to follow instructions because of coarse-grained evaluation per task not fine-grained instance-wise evaluation. MTEB (Muennighoff et al., 2022) is a similar type of benchmark that is widely used for evaluating instruction-tuned text embedding models, but it is not focused on retrieval tasks, so we do not compare it deeply. Additionally, M-BEIR (Wei et al., 2023) is introduced for assessing multimodal retrieval tasks but is also constrained by a task-description style.

3 The INSTRUCTIR Benchmark

3.1 Data Creation Pipeline

Constructing a framework to evaluate instruction-following capabilities in information retrieval models necessitates correlating multiple instructions with the same query and adjusting their targets accordingly (i.e., instruction, query, target text). Therefore, in contrast to previous approaches that evaluate coarse-grained task description-style instructions on information retrieval datasets with up to 15 instructions, we focus on creating per-query, instance-specific instructions as Table 1. We employ GPT-4 [2] (OpenAI, 2023) to facilitate the creation of such a setting. The development of our INSTRUCTIR evaluation datasets adheres to a systematic set of steps as Figure 2, outlined as follows:

Step 1. Select Seed Examples. In tackling the challenge of generating all components from scratch, we opt to levarage the MSMARCO dataset (Nguyen et al., 2016) for passage ranking into our seed examples [3]. This dataset is renowned for its comprehensive coverage of diverse topics collected from the real web. We carefully select a total of 1,743 queries Q and corresponding target texts T as seed examples, adhering to the following criteria: 1) The seed query should demonstrate the potential to match various targets as instructions evolve. 2) The target should offer substantial content, allowing for modifications that align with provided instructions. 3) Ensuring ease in controlling false negatives is crucial. To meet the first criterion, we focus on queries with a length ranging from 25% to 75% (i.e., 24 - 40). This approach prevents the seed query, pivotal for formulating subsequent instructions, from being overly vague or verbose. Similarly, for the second criterion, aligning with

[2] We use `gpt-4-1106-preview` for our work.

[3] We utilize the validation split from MSMARCO, consisting of 6,980 queries.

the rationale for query selection, we chose target texts within the text length range of 255 - 371 to avoid ambiguity. Lastly, for the third criterion, to facilitate the control of potential false negatives, we exclusively extract instances with only one positive target.

Step 2. Generate Instructions. Following the careful selection of seed examples in the previous step, we harness the power of GPT-4 to produce a set of instructions I_i corresponding to each query q_i, where i ranges from 1 to n, denoting the number of queries. To evaluate the adherence of retrievers to instructions, we emphasize the importance of deploying the same query with different instructions as illustrated in Figure 1. This approach enables us to measure how effectively models dynamically retrieve relevant targets. We also insist to move away from defining instructions in the form of rigid task descriptions, instead to embrace a more realistic approach that captures real-world scenarios in which retrieval systems are employed. This involves incorporating diverse information about users, such as their occupation, search context, location, search objectives, and preferred sources. To achieve this, we adopt the prompt outlined in Figure 4. We produce a set of instructions $I_i = \{I_{i,1}, ..., I_{i,k}\}$ (where k is set to 10) for each query q_i with a particular focus on aligning these instructions with a precise reflection of scenarios that real users may encounter.

Step 3. Revise Target Text. During this phase, GPT-4 refines the original target text T by integrating the instructions I_i generated in step 2. The model takes in a query q_i, an instruction $I_{i,k}$, and the original target t_i derived from seed examples. Subsequently, GPT-4 adjusts the target to better align with the provided instructions, resulting in $t^{'}_{i,k}$. This adjustment involves revising the target to accurately represent the given scenario, taking into account factors like the user's background, situation, location, occupation, hobbies, interests, or goals for the search. Additionally, it is crucial to incorporate information related to the user's preferences. To ensure the generation of diverse targets in response to evolving instructions, we employ a prompt, illustrated in Figure 5.

Step 4. Filtering Process. To assess the quality of machine-generated datasets during steps 2 and 3, we proceed by filtering out datasets in this stage. The selection of high-quality instances is based on the evaluation of two key criteria: *Q1. Does the revised target align with the original query?* and *Q2. Does the revised target align with the given query and instructions, while other targets do not?*. Additionally, we leverage the capabilities of GPT-4 as a quality evaluator. For Q1, we employ the prompt presented in Figure 7 to retain instances with a high score between the original query q_i and the revised target text $t^{'}_{i,k}$ [4]. To address Q2, the prompt in Figure 8 is utilized to identify instances where GPT-4 accurately predicts the gold target $t^{'}_{i,k}$ among distractors $t^{'}_{i,m}$ (where $m \neq k$), generated from the same query with different instructions. This is based on the scores between the original query q_i and an instruction $I_{i,k}$ pairs, and the set of revised target text $T^{'}_{i}$. Following the filtering stage, we select instances that possess more than 6 instructions with the same query, facilitating an effective evaluation of retrievers in dynamically changing scenarios. This results in a total of 9,906 instances, as detailed in Table 2. Further statistics regarding the filtering stage and examples can be found in the Appendix A.

	User-aligned	Type of inst.	# of inst.	Metrics
INSTRUCTIR	✓	instance-wise	9,906	Robustness, nDCG
BEIR	✗	task-wise	15	nDCG
LoTTE	✗	task-wise	5	nDCG
X2	✗	task-wise	6	nDCG

Table 1: Table comparing INSTRUCTIR with other IR benchmarks used to measure instruction following ability. Since the other benchmarks are not designed to evaluate instruction-following ability of information retrieval task, they are based on reformulated versions from previous studies (Asai et al., 2023; Su et al., 2022).

3.2 Dataset Analysis

Comparison Table. We characterize our INSTRUCTIR dataset as shown in Table 1. Note that all other benchmarks (BEIR, LoTTE, and X2) are not proposed for evaluating instruction following ability of retrieval systems originally, it is reformulated with author proposed instructions in the previous studies (Asai et al., 2023; Su et al., 2022; Wang et al., 2023). Other benchmarks are not focused on user aligned scenario, leading to structured formats that are far from search scenario of real users. Also, others utilize task-specific instructions, where up to 15 instructions are used for the evaluation, which

[4] Empirically, we observe that a score exceeding 3 out of 5 indicates good quality.

	Number
Avg. instructions per query	7.81
number of queries	1,267
Number of query with instructions	9,906
number of corpus	16,072
relevancy	binary

Table 2: Statitistics for InstrucIR dataset

is too small to evaluate whether a retrieval model can follow the instructions.

Dataset Quality. To ensure the quality and authenticity of our generated and filtered datasets, a human verification stage is implemented for validation purposes. For about 8% of the randomly sampled groups, a total of 10 annotators evaluate two instances per group. For each instance, we pose three questions, and three annotators are assigned to assess them, aiming to measure the inter-agreement between annotators (Cohen, 1960; Randolph, 2005). We consider the final human decision by majority voting. Further detail settings and results with inter-agreement are available in Appendix B. The first question (*Q1: Is the instruction valid for the search user?*) asks whether the provided instruction is suitable for the search user scenario. For this question, all instances were answered suitable for the search scenario. The second question (*Q2: Is the instruction natural for the given query?*) asks whether the instruction is naturally aligned with the query. About 97% instances were answered that the instruction and query have well-aligned relations. The third question (*Q3: Which passage is the most natural for the given instruction and query?*) asks to choose the most relevant passage. This question can estimate the difficulty of this benchmark, and evaluate the alignment between the results of Q2 from the filtering process (step 4) and human judgment. As a result, we got kappa coefficient (Cohen, 1960) of 0.6468 which indicates substantial agreement between human judgement and dataset (Landis and Koch, 1977). However, annotators demonstrate an approximately 76.5% top-1 accuracy, emphasizing the need for careful consideration when selecting the top-1 passage that follows the instructions. User interface and instruction for annotators are described in Figure 9.

Dataset Diversity and Statistics. Table 2 presents the data statistics for datasets within INSTRUCTIR. Through our data creation pipeline, we acquire an average of 7.81 instructions per query. Furthermore, to simulate real-world noisy search scenarios, we incorporate a subset of targets from the seed datasets. This results in an augmentation of approximately 6k additional targets, supplementing our revised targets. Consequently, our benchmark comprises a total of 10k instances, featuring a rich variety of instructions and a corpus with 16k entries, collectively constituting the INSTRUCTIR benchmark. Recognizing the significance of encompassing diverse instructions for the same query to reflect various user search scenarios, we evaluate the diversity of our dataset using the ROUGE score distribution inspired by Wang et al. (2022a). Specifically, we calculate the average ROUGE-L score for instructions associated with the same query. As illustrated in Figure 10, the instructions within INSTRUCTIR encapsulate highly diverse scenarios with low similarity to each other, achieving an average ROUGE-L score of 0.238 within each group.

3.3 Evaluation Metric

We employ the Normalized Cumulative Discount Gain (nDCG@k) following Thakur et al. (2021), as it is a widely accepted evaluation metric for assessing retrieval models. However, we emphasize the need for specialized metrics to assess the ability to follow instructions when retrieving information, especially when measuring dynamic changes in retrieval targets as instructions for a given query change. In instances where diverse search scenarios and user preferences serve as instructions alongside a given query, retrieval systems should adapt to these instructions to identify appropriate targets effectively. Hence, inspired by Zhong et al. (2022) and Oh et al. (2023), we introduce a Robustness score to assess how consistently the model predict targets over evolving instructions using the same query. To quantify robustness, we group instances with identical queries, calculate the minimum nDCG@k score within each group, and subsequently average these per-group scores to derive the final Robustness@k score.

4 Experiments

4.1 Baselines

We utilize INSTRUCTIR to compare various retriever systems in a zero-shot manner. INSTRUCTIR exclusively offers test datasets designed to evaluate the proficiency of existing retrievers in ad-

Instruction-tuned	Type	Models	Size	Robustness@10	nDCG@10
No	Lexical	BM25	-	26.92	76.01
	Late-Interaction	ColBERT-v2.0	110M	14.15	68.47
	Bi-Enc.	Contriever-msmarco	110M	47.40	84.85
		GTR-base	110M	34.06	73.35
		GTR-large	335M	37.56	75.95
		GTR-XL	1.5B	38.34	75.20
		RepLLaMa	7B	52.58	**87.62**
Yes	Bi-Enc.	TART-dual	110M	47.46	84.81
		INSTRUCTOR-base	110M	23.73	50.44
		INSTRUCTOR-large	335M	22.08	48.80
		INSTRUCTOR-XL	1.5B	21.53	48.63
		E5-mistral-7b-instruct	7B	**55.42**	86.33

Table 3: Zero-shot performances on INSTRUCTIR benchmark. We group models based on whether instruction-tuned or not, type of scoring methods, and the size of the models. Bi-Encoder is abbreviated as Bi-Enc.

dressing information retrieval tasks that require adherence to specific instructions. In our experimentation, we use pre-trained checkpoints accessible online for all models. We categorize the models based on the following criteria: non-instruction-tuned retrievers and instruction-tuned retrievers.

Non-instruction-tuned Retrievers. For non-instruction-tuned retrievers, we select BM25 (Robertson et al., 2009), Contriever-MSMARCO (Izacard et al., 2021), GTR (Ni et al., 2022), and RepLLaMa (Ma et al., 2023b). BM25 represents a lexical matching retriever. Contriever-MSMARCO, a bert-base sized model, is fine-tuned on the MSMARCO (Nguyen et al., 2016) passage ranking dataset following an unsupervised contrastive pre-training stage. GTR comprises variants of a t5-encoder based bi-encoder, trained on both MSMARCO and NQ datasets. RepLLaMa is a bi-encoder model based on the LLaMa-2-7b decoder, extracting token embeddings from the end-of-sequence token to generate text embeddings.

Instruction-tuned Retrievers. For instruction-tuned retrievers, we utilize TART-dual (Asai et al., 2023), INSTRUCTOR (Su et al., 2022), and E5-mistral-7b-instruct (Wang et al., 2023). TART-dual is a retriever with a bi-encoder architecture based on the Contriever model. It is additionally trained on BERRI, an instruction-aware information retrieval dataset comprising approximately 40 diverse retrieval tasks, each accompanied by a task description serving as instructions. INSTRUCTOR offers various size versions, including base, large, and xl, all finetuned on the GTR retriever as a backbone [5]. It is further trained on the MEDI dataset, encompassing 330 distinct text embedding tasks, each with a human-written task instruction, including multiple retrieval datasets. Lastly, E5-mistral-7b-instruct is a bi-encoder architecture retriever trained on a proprietary Language Model(LLM), the mistral-7b decoder. It undergoes training solely with synthetic data for text embedding tasks, with task definitions serving as instructions.

4.2 Results

We evaluate various retrieval systems on INSTRUCTIR benchmark to evaluate their capability to follow instructions in a zero-shot setting as Table 3. The baselines are categorized into instruction-tuned and non-instruction-tuned models across different sizes and architectures.

Non-Instruction-Tuned Models. Considering overall high score of nDCG over 50, due to the characteristics of distinct instructions in INSTRUCTIR datasets, lexical bias exists. However, the lexical matching model BM25 and late interaction model ColBERT-v2.0 show Robustness@10 of 26.92 and 14.15 respectively. And it means that focusing individual keywords can't truly understand evolving instructions, which leads to huge drop in Robustness score. Representative retriev-

[5] As INSTRUCTOR also leverages corpus-side instruction, we adhere to its approach by using the corpus instruction: '*Represent the document for retrieval:*' utilized for the MS-MARCO dataset.

ers that are trained with contrastive training objectives, such as Contriever-msmarco and GTR variants (base, large, and xl), are considered, with sizes ranging from 110M to 1.5B parameters. These models exhibited a significant improvement in Robustness over BM25, which Contriever-msmarco shows most powerful and robust performance compared to the same size models in non-instruction tuned baselines even superior to way more larger size GTR-xl. The largest model, RepLLaMa with 7B parameters, achieve the highest nDCG@10 of 87.62 and Robustness@10 of 52.58, indicating a strong correlation between model size and performance metrics in non-instruction-tuned settings.

Instruction-Tuned Models. TART-dual undergoes additional training steps with BERRI, a comprehensive set of over 40 retrieval datasets accompanied by task-specific instructions generated by human experts, based on Contriever-msmarco. INSTRUCTOR variants are subsequently trained, aligning with the respective sizes of GTR retrievers, using MEDI datasets. These datasets are reformulated training sets covering 330 datasets, incorporating instructions spanning diverse text embedding task categories and domains, including retrieval tasks. However, both series of instruction-tuned baselines do not show superior performance than their non-instruction-tuned counterparts. Notably, INSTRUCTOR variants exhibit a huge performance drop compared to the backbone model GTR variants. This can be interpreted that finetuning retrievers with only task-style instructions doesn't guarantee good performance in various free creative user-aligned style instructions. Conversely, E5-mistral-7b-instruct, shows Robustness@10 with 55.42 outperforming all other models. This highlights the importance of using large instruction-tuned models for search tasks to follow instructions as well.

5 Analysis

Scaling Up Model Size Leads Better Instruction Following. It is not surprising that larger models derive greater benefits from instruction tuning, indicating that their enhanced capacity enables a more effective integration of instruction-following abilities. In the case of non-instruction-tuned baselines, GTR demonstrates superior performance as model sizes increase, particularly in terms of Robustness scores. Remarkably, Contriever-msmarco exhibits competitive performance even with smaller sizes. However, INSTRUCTOR variants display reverse trends, resulting in lower Robustness scores as model sizes increase. This hints at an overfitting issue to diverse task description style instructions, leading to diminished performance across varying instruction styles and longer, unseen user-aligned scenarios. Nonetheless, the outstanding performances of 7B size models, RepLLaMA, and E5-mitral-7b-instruct underscore the significance of both model size and instruction tuning in developing competent retrieval systems for complex, instruction-based queries.

Models	Robustness Gap	nDCG Gap
BM25	-17.62	-14.69
Contriever-msmarco	-2.31	-1.78
ColBERT-v2.0	+22.20	+10.45
GTR-base	-2.62	-1.53
GTR-large	-1.97	-1.34
GTR-XL	-2.53	-1.22
RepLLaMa	-4.11	-2.94
TART-dual	-0.58	-0.82
INSTRUCTOR-base	+ 17.06	+ 32.82
INSTRUCTOR-large	**+ 34.70**	**+ 40.74**
INSTRUCTOR-XL	+23.26	+ 37.27
E5-mistral-7b-instruct	+2.89	+1.61

Table 4: Performance gap for changing instruction order.

Instruction Order Sensitivity Exists For Instruction Tuned Retrievers. To analyze the importance of the order of instructions and queries, we conduct additional experiments by changing their sequence (query-> instruction). Remarkably, INSTRUCTOR exhibits a significant performance gain when query precede the instructions as Table 4. Considering the average length of instructions in the training dataset used for INSTRUCTOR, it is approximately 12.16 tokens. In contrast, the average number of tokens for instructions utilized in our user-aligned instructions is about 64.47. This discrepancy highlights the challenge of generalizing models trained solely on task-description style instructions, which exhibit limited creativity and variety, to more user-aligned cases. For a detailed comparison between coarse-grained task description instructions and user-aligned instance-specific instructions, please refer to the examples provided in the Appendix 8.

Weighting Individual Terms leads Instruction Sensitivity for the Paraphrased Instructions. To analyze the sensitivity of instructions when paraphrased, we randomly select one instance from

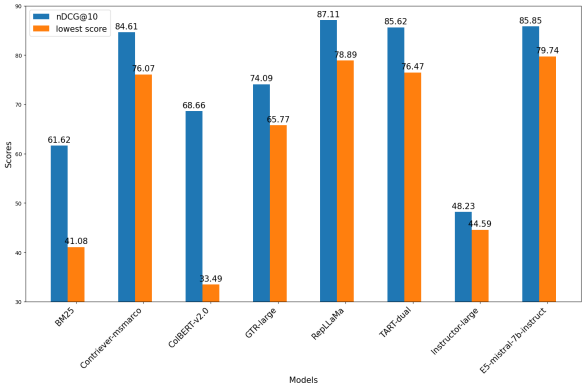

Figure 3: Prompt sensitivity per models. Blue bar and orange bar denote performance of original instructions and smallest score of paraphrased versions respectively.

Models	Robustness Gap	nDCG Gap
Contriever-msmarco	+4.10	-25.31
ColBERT-v2.0	**+10.91**	**-36.43**
GTR-base	+3.09	-27.95
GTR-large	+5.95	-24.48
GTR-XL	+4.65	-24.55
RepLLaMa	+1.35	-24.96
TART-dual	+6.57	-22.93
INSTRUCTOR-base	+6.65	-14.73
INSTRUCTOR-large	+9.62	-11.80
INSTRUCTOR-XL	+10.74	-11.36
E5-mistral-7b-instruct	+6.90	-17.14

Table 5: Performance gap when removing high lexical overlap. BM25 is removed as used for filtering.

each group in INSTRUCTIR, resulting in 1,267 subsets of instances. Subsequently, we generate five paraphrased versions for each instance using GPT-4. This process yields a total of 1,267 groups of paraphrased instructions. The evaluation is based on the smallest score among the five paraphrased instructions. As illustrated in Figure 3, retrievers specialized in emphasizing specific terms, such as BM25 and ColBERT-v2.0, exhibit a substantial performance drop of 20.53 and 35.17, respectively. In contrast, most bi-encoder based models demonstrate less fluctuation compared to these two models. Notably, the E5-mistral-7b-instruct model displays the most robust performance in adapting to changing instructions.

Relying on Lexical Redundancy Reduces Robustness. Considering existence of distinct information per the instruction, lexical hint can exist to find proper target. Therefore, we eliminate instances with high lexical overlap by filtering out cases where BM25 predicts the ground truth answer in the top 10, and evaluate how other semantic matching models are affected by this filtering process. Overall, average Robustness score for the less lexical overlap cases increases in all model, and nDCG score drops. And it means some of datasets in INSTRUCTIR can be solved by lexical matching, but relying heavily on lexical cues can lead to wrong targets, which leads to lower Robustness score. Among all, ColBERT-v2.0 shows the largest changes showing average plus 10.91 Robustness score, and minus 36.43 nDCG score. RepLLaMA shows least Robustness performance change.

6 Conclusion and Future Work

In this study we introduce a novel benchmark, INSTRUCTIR, designed for evaluating the instruction following capabilities of information retrieval models. Despite the critical importance of aligning models with user instructions and reflecting user preferences in information retrieval tasks, existing evaluations often fall short in comprehensively assessing these aspects. Therefore, our benchmark focuses on evaluating user-aligned instructions tailored to each query instance, reflecting the diverse characteristics inherent in real-world search scenarios. Our experimental investigation sheds light on the instruction following capabilities of information retrieval models, presenting valuable insights that contribute to the current understanding of this domain. In particular, we highlight the gaps in current instruction-tuned retrievers in the limited style of training datasets organized around task description style instructions.

One promising direction for future research is the exploration of methodologies, such as Reinforcement Learning from Human Feedback (RLHF), to enhance the alignment of retrieval models with users' search intentions as proposed by Ouyang et al. (2022). Investigating the integration of RLHF techniques can potentially lead to more effective and adaptive information retrieval systems that better understand and respond to user instructions. Additionally, future studies could delve into the development of more diverse instruction-aware retrieval training datasets that capture the nuances of user preferences and instructions in a more intricate manner. By addressing these challenges, we anticipate significant advancements in the field, ultimately improving the overall user experience in information retrieval scenarios.

References

Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. 2023. Task-aware retrieval with instructions. In *Findings of the ACL*.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*.

Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback. *arXiv*.

Avia Efrat, Or Honovich, and Omer Levy. 2022. Lmentry: A language model benchmark of elementary language tasks. *arXiv*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *Trans. Mach. Learn. Res.*

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2023. Followbench: A multi-level fine-grained constraints following benchmark for large language models. *arXiv*.

J Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv*.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023a. Query rewriting for retrieval-augmented large language models. *arXiv*.

Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2023b. Fine-tuning llama for multi-stage text retrieval. *arXiv*.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2022. Mteb: Massive text embedding benchmark. In *EACL*.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human-generated machine reading comprehension dataset.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, et al. 2022. Large dual encoders are generalizable retrievers. In *EMNLP*.

Hanseok Oh, Haebin Shin, Miyoung Ko, Hyunji Lee, and Minjoon Seo. 2023. Ktrl+ f: Knowledge-augmented in-document search. *arXiv*.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv*.

Justus J. Randolph. 2005. Free-marginal multirater kappa (multirater k[free]): An alternative to fleiss' fixed-marginal multirater kappa.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Keshav Santhanam, O. Khattab, Jon Saad-Falcon, Christopher Potts, and Matei A. Zaharia. 2021. Colbertv2: Effective and efficient retrieval via lightweight late interaction. In *NAACL*.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. One embedder, any task: Instruction-finetuned text embeddings. *ArXiv*.

Nandan Thakur, Nils Reimers, Andreas Ruckl'e, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *ArXiv*.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2023. Improving text embeddings with large language models. *arXiv*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022a. Self-instruct: Aligning language model with self generated instructions. *arXiv*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022b. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *arXiv*.

Cong Wei, Yang Chen, Haonan Chen, Hexiang Hu, Ge Zhang, Jie Fu, Alan Ritter, and Wenhu Chen. 2023. Uniir: Training and benchmarking universal multimodal information retrievers. *ArXiv*.

Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. 2023. Flask: Fine-grained language model evaluation based on alignment skill sets. *arXiv*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2023a. Instruction tuning for large language models: A survey. *ArXiv*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023b. Instruction tuning for large language models: A survey. *arXiv*.

Victor Zhong, Weijia Shi, Wen-tau Yih, and Luke Zettlemoyer. 2022. Romqa: A benchmark for robust, multi-evidence, multi-answer question answering. *arXiv*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv*.

Appendix

A Details for Dataset Construction Pipeline

Filtering Process. During the filtering process in step 4 of data creation pipeline, we perform two filtering criteria. For the first criterion, *Q1. Does the revised target align with the original query?*, we utilize a prompt in Figure 7. To select high relevant revised target for the given query, we measure scores for the given score rubric from 1 to 5 and corresponding explanation. Average score distribution for this step is in Figure 6. When we randomly sample the outputs, score exceeding 3 out of 5 shows good quality. After this process, 15,669 instances are survived out of 16,157 instances. Detailed examples with both high and low scores are available in the Table 6. Next, for the second criterion, *Q2. Does the revised target align with the given query and instructions, while other targets do not?.*, we use a prompt in Figure 8. In this step, we only select correct case that GPT-4 predicts annotated target among other distractors, and 11,992 instances are selected out of 16,157. After merging these two steps, and select where more than 6 instances exist per group, we get total 1,267 groups with 9,906 instances left.

B Details for Human Verification

User Interface. Ten annotators are instructed to answer three questions as illusted in Figure 9. For the first question (*Q1: Is the instruction valid for the search user?*) and second question (*Q2: Is the instruction natural for the given query?*), annotators are required to choose either *Yes* or *No*. If the answer is *No*, annotator are also required to provide a short reason. In the third question (*Q3: Which passage is the most natural for the given instruction and query?*), annotators are provided three passages from the same group, including the correct answer.

Evaluation Settings. For each instance, the responses from three annotators are reported and regard the final human judgement by majority voting. In the first and second questions, we assess the reliability of the annotator's responses through kappa coefficient (Randolph, 2005) with the proportion of *Yes* responses. In the third question, we compare the final human judgement and INSTRUCTIR dataset through kappa coefficient (Cohen, 1960) and report the top-1 accuracy.

Inter-agreement. Following the Landis and Koch (1977), the kappa coefficient for each question is interpreted as follow: *poor agreement* (< 0); *slight agreement* (0.01-0.20); *fair agreement* (0.21–0.40); *moderate agreement* (0.41–0.60); *substantial agreement* (0.61–0.80); *almost perfect agreement* (0.81–1.00). The first question shows *almost perfect agreement*, and the second question shows *moderate agreement* among the three annotators for each instance. And the third question shows *substantial agreement* between the human judgement and INSTRUCTIR dataset.

Q1. Is the instruction valid for the search user?	
kappa coefficient (Randolph, 2005)	0.9133
Proportion of *Yes* response	100%
Q2. Is the instruction natural for the given query?	
kappa coefficient (Randolph, 2005)	0.5267
Proportion of *Yes* response	97%
Q3. Which passage is the most natural for the given instruction and query?	
kappa coefficient (Cohen, 1960)	0.6468
Top-1 Accuracy	76.5%

Table 7: Human verification results for INSTRUCTIR.

```
>> SYSTEM PROMPT
You are a helpful, respectful and creative assistant.

>> USER INSTRUCTION
Your task is to generate a set of scenarios for the provided search query.
Here is the specification for the scenario generation task:
- The scenario should reflect a very specific scenario where a user is interacting with an AI search engine.
- Within the scenario, the user could write about his/her job, background, situation, location, occupation,
    hobbies, interests, or goals of doing the search. Also, the user could explicitly reflect about
    his/her preference regarding the document to be searched.
- The scenario SHOULD be written from a first person's view point. For example, it should start with
    phrases like "I am a {job}, ...", "I am in a situation ...", "During my {situation}".
- While the provided query is about "what" is being searched for, the scenario you will generate should be
    about "how" the search should be approached and what values or criteria should be prioritized in that
    search. This distinction makes the role of the scenario clear as a guiding framework for the search
    process, as opposed to the query which is more about the specific target of the search.
- However, the scenario should be RELATED with the provided query. In other words, it shouldn't be
    applicable to other queries in general.
- You should generate based on the following format (note that there is a phrase
"[END]" after each scenario being generated):
Scenario 1: {generate the first scenario} [END]
Scenario 2: {generate the second scenario} [END]

Scenario 10: {generate the last scenario} [END]
- Please do not generate any other opening, closing, and explanations. Just generate the set of scenarios!
```

Figure 4: Prompt for generating instructions (step 2)

```
>> System Prompt
You are a helpful, respectful and creative assistant.

>> USER INSTRUCTION
Your task is to generate a REVISED DOCUMENT for the provided search QUERY and SCENARIO pair.
Here is the specification for the DOCUMENT revising task:
- The REVISED DOCUMENT should reflect the user's unique SCENARIO where a user is interacting with an AI
    search engine.
- Within the REVISED DOCUMENT, revise details reflecting the user's background, situation, location,
    occupation, hobbies, interests, or goals of doing the search. Also, containing information related to
    the user's preference is important.
- Directly revise given DOCUMENT that has good quality that can be found by an AI search engine. Don't just
    suggest it!
- Do NOT include the same keywords from the given SCENARIO in REVISED DOCUMENT. Paraphrase it.
- However, the REVISED DOCUMENT should be RELATED with the provided query. In other words, it should be
    applicable to query in general.
- You should generate based on the following format (note that there is a phrase "[END]" after each
    elements being generated):

PLAN: {generate the plan for the strategy for revision} [END]
REVISED DOCUMENT: {revise the document} [END]

- Please do not generate any other opening, closing, and explanations. Just generate the PLAN and REVISED
    TARGET !
```

Figure 5: Prompt for revising target text (step 3)

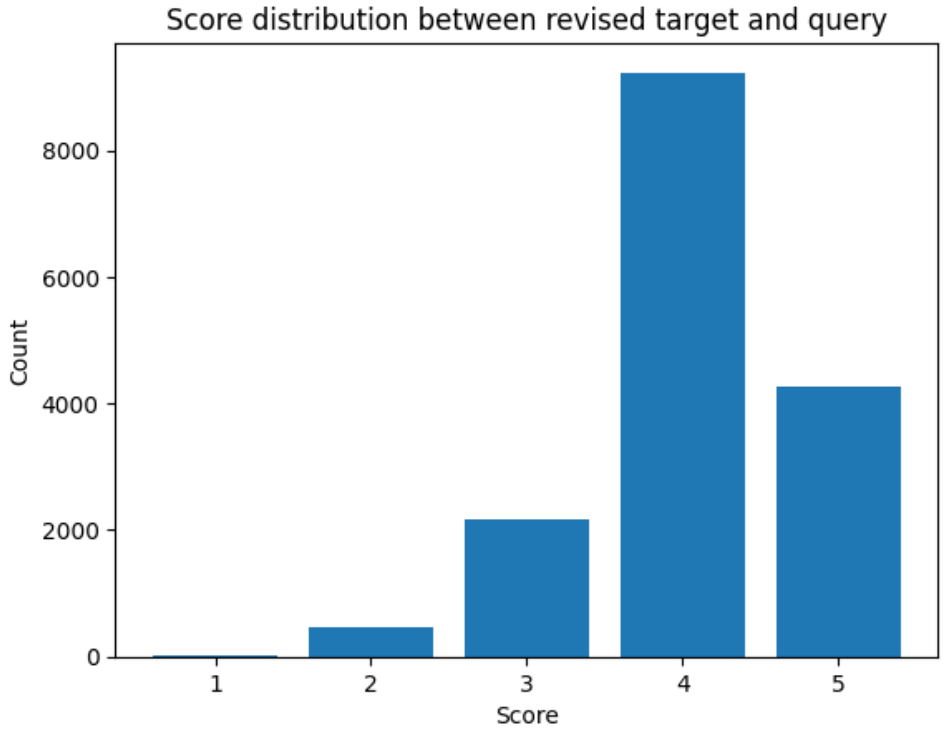

Figure 6: Relevancy score distribution between revised target after data generation step3 and query.

```
>>> SYSTEM PROMPT
You are a helpful, respectful and creative assistant.

>>> USER INSTRUCTION
You are a similarity evaluator! You're tasked with calculating the similarity between QUERY, and DOCUMENT
     displayed below based on their relevancy. In the evaluation, I want you to rate the relevancy of the
     pair according to the following score rubric:

Score 1: The DOCUMENT, QUERY have very little or no relevance to each other. The elements compared share
     almost no common attributes or context.
Score 2: The DOCUMENT, QUERY have some relevance but are quite distinct. They share a few attributes or
     contextual details, but there are significant differences in the majority of aspects.
Score 3: The DOCUMENT, QUERY are moderately relevant to each other. They share a fair amount of attributes
     or context, but there are still some notable differences that prevent a high similarity score.
Score 4: The DOCUMENT, QUERY have high relevance to each other. They share many attributes or contextual
     details, with only a few differences that do not majorly impact the overall similarity.
Score 5: The DOCUMENT, QUERY are very highly relevant or almost identical to each other. They share nearly
     all attributes or the context is almost exactly the same, with very minor or negligible differences.

You will be given QUERY and DOCUMENT pair.
[QUERY]
{{query}}

[DOCUMENT]
{{candidate document}}

You should generate based on the following format:

<Explanation>
{explanation for the score}
</Explanation>

<Score>
{score}
</Score>

Please give feedback on the DOCUMENT with respect to each QUERY, and provide a score on a scale of 1 to 5
     whether it satisfies the requirements, where a higher score indicates better performance.
```

Figure 7: Prompt for filtering stage Q1 (step 4).

Type	Example
Good	**[Query]** Heteromorphic definition. **[Target]** In the realm of genetic research and study, "heteromorphic" specifically denotes the occurrence of distinct forms, or morphs, within an organism's life cycle, or among different organisms, as a result of genetic mutation. This term is often utilized in scholarly articles to describe variations that are not merely different in size but also morphologically distinct, such as varying chromosome shapes in the gametes of different sexes of the same species. In an educational setting, one would explore the significance of heteromorphism in understanding evolutionary processes and the resulting diversity among living forms. It's applied in studies that examine the genetic basis of these morphological variations and their implications for the survival and reproduction of species. **[Score]** 5 **[Explanation]** The document provides a detailed explanation of the term "heteromorphic" in the context of genetic research and study, explaining that it refers to the occurrence of distinct forms within an organism's life cycle or among different organisms as a result of genetic mutation. It also discusses the relevance of heteromorphism in evolutionary processes and the diversity of living forms. This explanation aligns well with the query, which asks for the definition of "heteromorphic." The document offers a clear, in-depth definition and contextualizes the term within its scientific application, demonstrating a high relevance to the query.
Bad	**[Query]** is catecholamine a steroid **[Target]** For those actively engaged in physical conditioning, it's crucial to understand the role of various hormones. In the body's natural arsenal, there are hormones derived from the amino acid tyrosine that have significant roles, such as the thyroid hormone, which is pivotal for organ development and metabolic rate—an aspect vital for anyone with a focus on fitness. **[Score]** 1 **[Explanation]** the query is asking whether catecholamine is a steroid, which is a question about categorization of a substance based on its chemical nature. the document discusses hormones derived from the amino acid tyrosine, mentioning thyroid hormone and its role in metabolism and organ development, which is related to the broader topic of hormones. however, the document does not specifically address catecholamines or steroids, nor does it provide information to determine whether catecholamines are steroids. therefore, while there is a connection in the general subject matter (hormones), the specific question posed by the query is not answered by the document.

Table 6: Dataset example for filtering stage Q1 (step 4).

```
>>> SYSTEM PROMPT
You are a helpful, respectful and creative assistant.

>>> USER INSTRUCTION
You are a ranker agent! Each potential DOCUMENT has a corresponding DOCUMENT id and you're tasked with
     ranking the answers based on their relevancy to the pair of QUERY, SCENARIO pair. In the evaluation, I
     want you to rate the relevancy of the pair according to the following score rubric:

Score 1: The DOCUMENT lacks relevance to the user's SCENARIO, providing little to no connection to the
     user's job, background, situation, location, occupation, hobbies, interests, or goals. It fails to
     consider preferences and context, resulting in an overall inadequate fit.
Score 2: The DOCUMENT has limited relevance, with only a few elements aligning with the user's SCENARIO and
     QUERY. While some contextual understanding and preference consideration may be present, it falls short
     of providing a comprehensive and well-fitted response.
Score 3: The DOCUMENT demonstrates moderate relevance, capturing some aspects of the user's SCENARIO. It
     shows an adequate contextual fit and considers a majority of the user's stated preferences. However,
     there is room for improvement in terms of depth and clarity.
Score 4: The DOCUMENT exhibits high relevance, aligning well with the user's SCENARIO and covering most
     relevant aspects. It demonstrates a strong contextual fit, addresses the user's preferences
     effectively, and maintains high clarity and conciseness. However, there may be minor areas for
     improvement.
Score 5: The DOCUMENT is perfectly relevant, precisely addressing all aspects of the user's SCENARIO,
     QUERY, and preferences. It seamlessly integrates with the user's context, demonstrating a profound
     understanding. The DOCUMENT is exceptionally clear, concise, and exhaustive in providing information,
     offering a flawless fit.

You SHOULD ONLY generate the top ranked id from the given search DOCUMENT (id : 1~10) and no additional
     comments as [id]. This is VERY IMPORTANT!

You will be given list of DOCUMENT and a pair of QUERY,SCENARIO.

[DOCUMENT LIST]
[1] {DOCUMENT_1}
[2] {DOCUMENT_2}
[3] {DOCUMENT_3}
...
[10] {DOCUMENT_10}

[SCENARIO]
{scenario}

[QUERY]
{query}

You should generate based on the following format:
<Explanation>
{explanation for the ranking}
</Explanation>

<Ranking>
{top ranked DOCUMENT id}
</Ranking>

Please give a top ranked DOCUMENT id with respect to each SCENARIO and QUERY pair, and provide a score on a
     scale of 1 to 5 whether it satisfies the requirements, where a higher score indicates better
     performance.
```

Figure 8: Prompt for filtering stage Q2 (step 4).

Project Goal

We aim to construct a benchmark focusing on instance-specific and user-aligned instructions for each query, mirroring the varied nature of real-world search scenarios.

Instructions for Data Labeler

You will encounter three questions for each instance. Your task is to choose an option for each question. **If you choose 'NO' for Q1 or Q2, provide a clear and concise reason for your decision.**

About Questions

1. Q1 asks whether the provided instruction is suitable for the search user scenario.

- **VALID**: providing information about the search user, such as the user's job, background, situation, location, occupation, hobbies, interests, goals or preferred targets.
- **INVALID**: providing little to no connection to the user's job, background, situation, location, occupation, hobbies, interests, or goals. It fails to consider preferences and context, resulting in an overall inadequate fit.

2. Q2 asks whether the given instruction aligns naturally with the provided query.

- **NATURAL**: The scenario reflects a very specific scenario in which a user interacts with a search engine using a given query and typically instance specific and not universally applicable to other queries.
- **UNNATURAL**: The scenario does not provide any additional information that is not processed in the query. Throughout, redundant information and overly general scenarios are provided.

3. Q3 is asking to identify the passage that best fits the given instruction and query, considering specific conditions such as occupation, age, situation, and preferences. Pay careful attention to detail when selecting the most appropriate target from the provided options.

- **NOT RELEVANT**: The passage lacks relevance to the user's INSTRUCTION, providing little to no connection to the user's job, background, situation, location, occupation, hobbies, interests, or goals. It fails to consider preferences and context, resulting in an overall inadequate fit.
- **RELEVANT**: The passage is mostly relevant, precisely addressing all aspects of the user's INSTRUCTION, QUERY, and preferences. It seamlessly integrates with the user's context, demonstrating a profound understanding. The passage is exceptionally clear, concise, and exhaustive in providing information, offering a flawless fit.

(a) Instructions for Annotators.

Instruction

I am a health policy analyst, and I'm tasked with evaluating the efficiency of endocrinology clinics in rural settings. My search should focus on studies or reports that compare various clinic models and their outcomes in rural versus urban areas.

Query

what is endocrinology clinic

Q1. Is the ***instruction*** valid for the search user?

☑ Yes[1] ☐ No[2]

Q2. Is the ***instruction*** natural for the given ***query***?

☑ Yes[3] ☐ No[4]

Q3. Which passage is the most natural for the given ***instruction*** and ***query***?

☐ An endocrinology clinic specializes in the disease management and therapeutic interventions involving hormonal disorders. In establishing a top-tier endocrinological department, it's crucial to examine comprehensive evaluations from distinguished institutions. This includes analyzing how these clinics uphold medical protocols. Adept knowledge is essential in managing treatments that affect multiple organ systems, embodying a holistic approach to patient care. Example reports would detail operational frameworks, patient outcomes, and how interdisciplinary teams contribute to cutting-edge endocrine healthcare.

[5]

☐ An endocrinology clinic focuses on glandular disorders and the wide-ranging effects of hormones in the human body. These specialized medical centers frequently employ sophisticated electronic health systems to streamline patient information, scheduling, and treatment plans. Integral to their operation is software that can manage detailed hormone therapy regimens and track patient progress with precision. The technology adopted in these clinics enables medical professionals to effectively coordinate care, monitor metabolic responses, and deliver personalized therapy for chronic conditions like diabetes or thyroid disorders. Medical IT solutions for these practices are tailored to support a multidisciplinary approach that encompasses diverse aspects of patient care, including data analysis, medication management, and ongoing communication with primary care providers.

[6]

☑ Endocrinology clinics are crucial healthcare facilities that specialize in diagnosing and treating hormonal imbalances and issues related to the endocrine glands. The effectiveness of these clinics can markedly differ between rural and urban locales due to varying challenges, such as access to advanced medical technologies and expertise. Evaluative reports have highlighted that although urban clinics typically have access to a broader range of resources and a higher patient volume, rural clinics play a critical role in providing specialized care in less dense populations. Such studies underscore the importance of assessing how varying models of clinic operations impact patient outcomes and service delivery in different geographical settings, paving the way for policy development aimed at optimizing healthcare delivery for endocrine disorders across diverse clinical environments.

[7]

(b) Questions for each instance.

Figure 9: User Interface for Human Verification.

[Query] ***what is the annual fee for spirit***

[Instruction]

I am a budget-conscious traveler looking to cut costs on my frequent trips. I prefer to invest more in experiences than in transport, so I need accurate information on the annual fee for Spirit Airlines' membership or credit card programs that offer benefits like waived bag fees or discounts.

[Target]

Thrifty explorers who habitually travel can find value in Spirit's credit card, especially since the $59 fee is waived for the initial 12 months, allowing for savings early on. Although to offset the cost from the second year onwards, cardholders are expected to spend at least $5,900 annually. Yet, the card can be lucrative for those who accumulate enough expenses, as it offers benefits like complimentary bag exemptions and fare reductions—perks that align with the preference to allot more for enriching experiences over transport expenses.

[Instruction]

As a travel agent, I'm preparing a cost-analysis presentation for a client who's interested in low-cost carriers. I am searching for the most up-to-date annual fee for Spirit's exclusive membership to include in my comparison chart alongside other budget airline fees.

[Target]

The exclusive membership offered by Spirit, recognized for its affordability, incurs an annual fee of $59. This figure is essential for clients evaluating the comparative costs of low-cost airlines and determining the best value for their frequent travels. Notably, this fee is typically waived during the initial year of membership, enhancing the cost-effectiveness for new members. The fee is a crucial datapoint in the broader context of budget travel expenses and should be accounted for when mapping out annual travel budgets for cost-conscious clients.

[Instruction]

I am a college student on a tight budget with a penchant for exploring new places. I am researching various airlines' membership fees, particularly Spirit's annual fee, to figure out if the cost-saving benefits align with my limited resources and travel frequency.

[Target]

Balancing a wallet-friendly lifestyle while nurturing your wanderlust can be challenging, especially when it comes to airline memberships. Taking a closer look at Spirit's offerings, there's a $59 yearly charge for their credit card, fortunately waived for the initial twelve months. To offset this expense, an expenditure of $5,900 on the card annually is needed. One should consider their usual spending habits and the practicality of reaching such a sum in regular yearly outlays to determine if the membership aligns with prudent financial management as a diligent academic explorer.

[Instruction]

I am a Human Resources professional planning the company's travel budget for the upcoming fiscal year. I need to find the annual fee for Spirit to calculate whether bulk memberships for our staff travel would be advantageous compared to pay-per-use options.

[Target]

The annual membership fee for Spirit's corporate clients is designed to accommodate businesses looking to manage their travel expenses more effectively. While individual memberships or pay-per-use arrangements can potentially add to travel costs over time, opting for an annual corporate membership can offer substantial savings. This approach ensures that a company's travel budget is maximized, especially when considering the volume of staff travel. Keep in mind that corporate packages may offer additional perks, such as discounts on group bookings and other travel-related services, which further enhance the value proposition when compared to transaction-based fees. Evaluating these membership fees and aligning them with the company's projected travel volume will determine the most cost-efficient strategy for the fiscal year's travel budget.

Table 8: Dataset example in INSTRUCTIR benchmark.

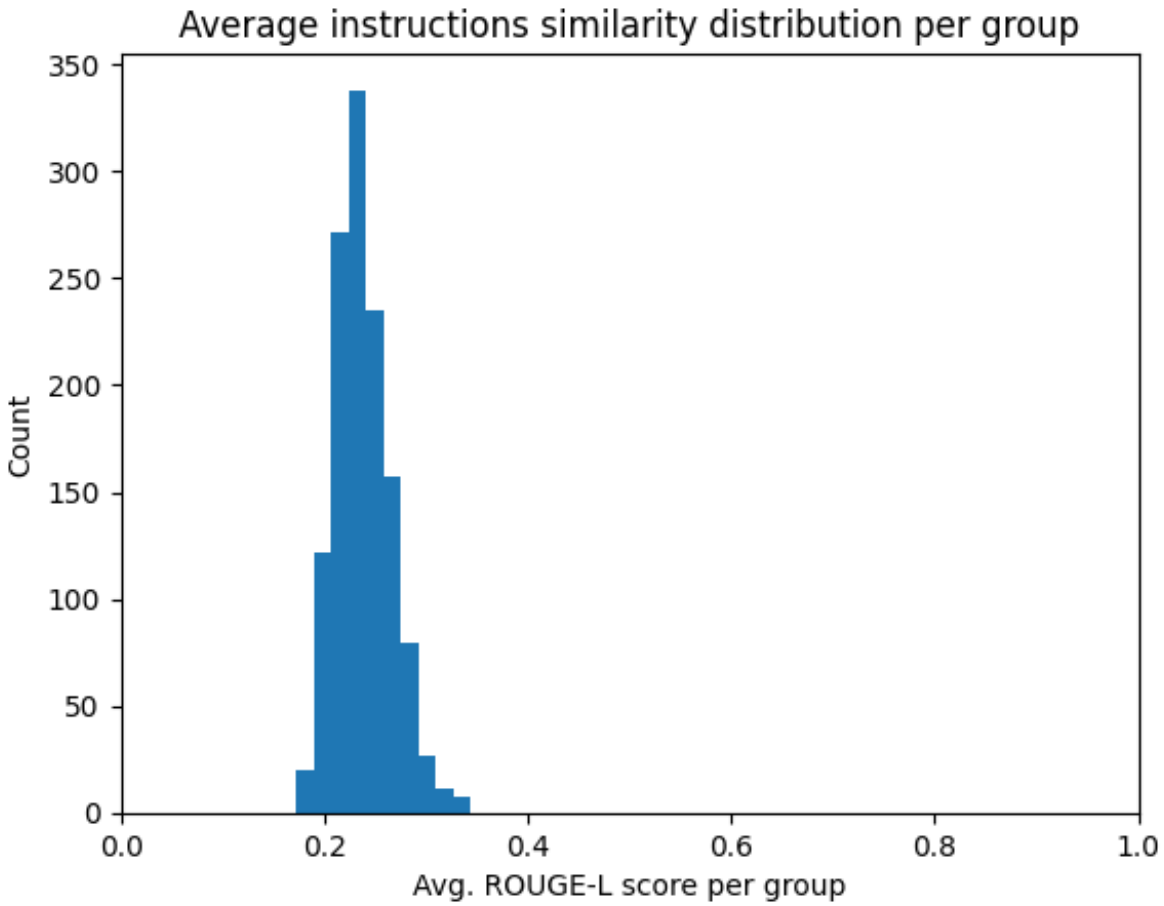

Figure 10: ROUGE-L score distribution for the instructions.