# MIRAGE-BENCH: Automatic Multilingual Benchmark Arena for Retrieval-Augmented Generation Systems

Nandan Thakur<sup>1</sup>, Suleman Kazi<sup>2</sup>, Ge Luo<sup>2</sup>, Jimmy Lin<sup>1</sup>, Amin Ahmad<sup>2</sup>

<sup>1</sup> University of Waterloo, Canada <sup>2</sup> Vectara, USA

{nandan.thakur,jimmylin}@uwaterloo.ca
 {suleman,rogger,amin}@vectara.com

#### **Abstract**

Traditional Retrieval-Augmented Generation (RAG) benchmarks rely on different heuristicbased metrics for evaluation, but these require human preferences as ground truth for reference. In contrast, arena-based benchmarks, where two models compete each other, require an expensive Large Language Model (LLM) as a judge for a reliable evaluation. We present an easy and efficient technique to get the best of both worlds. The idea is to train a learning to rank model as a "surrogate" judge using RAGbased evaluation heuristics as input, to produce a synthetic arena-based leaderboard. Using this idea, We develop MIRAGE-BENCH, a standardized arena-based multilingual RAG benchmark for 18 diverse languages on Wikipedia. The benchmark is constructed using MIRACL, a retrieval dataset, and extended for multilingual generation evaluation. MIRAGE-BENCH evaluates RAG extensively coupling both heuristic features and LLM as a judge evaluator. In our work, we benchmark 19 diverse multilingualfocused LLMs, and achieve a high correlation (Kendall Tau ( $\tau$ ) = 0.909) using our surrogate judge learned using heuristic features with pairwise evaluations and between GPT-40 as a teacher on MIRAGE-BENCH leaderboard using the Bradley-Terry framework. We observe proprietary and large open-source LLMs currently dominate in multilingual RAG. MI-RAGE-BENCH is available: https://github. com/vectara/mirage-bench.1

### 1 Introduction

Large Language Models (LLMs) have recently gained popularity for information-seeking queries leading to the widespread adoption of Retrieval-Augmented Generation (RAG) (Guu et al., 2020; Lewis et al., 2020; Izacard and Grave, 2021; Borgeaud et al., 2022). The naive RAG setup traditionally includes a retrieval and a generation

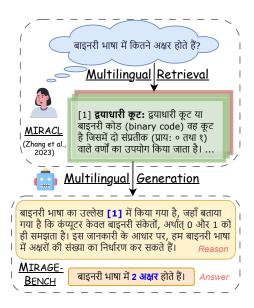


Figure 1: Multilingual naive RAG pipeline in Hindi (hn). In MIRAGE-BENCH, we reuse the oracle retrieval set (query, judged relevant and non-relevant passages) in MIRACL (Zhang et al., 2023), and extend evaluation for the multilingual answer generation stage with LLMs.

stage, conducted sequentially. RAG systems such as Bing Search (Microsoft, 2023) provide grounded responses, i.e., statments include citations to one or more retrieved passages. The citations reduce factual hallucinations with easy verifiability and improve support or faithfulness to passages provided within context (Khandelwal et al., 2020; Lewis et al., 2020; Gao et al., 2023a,b; Liu et al., 2024). However, existing RAG benchmarks are Englishcentric, due to uneven and scarce data available across multiple languages (Thakur et al., 2024b). So far, it is unclear how well existing LLMs perform in multilingual RAG, i.e., where queries and passages are non-English and the LLM generates a response in the same language. An example of RAG in Hindi language (hn) is shown in Figure 1.

Existing RAG benchmarks can be broadly classified as either (i) *heuristic-based*, where benchmarks design multiple evaluation metrics to evaluate systems across multiple dimensions (Gao et al.,

<sup>&</sup>lt;sup>1</sup>MIRAGE-BENCH has been coined as ( $\underline{MI}$ RACL +  $\underline{RAG}$  +  $\underline{E}$ VALUATION +  $\underline{BENCH}$ MARK).

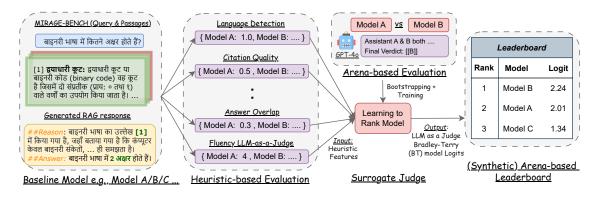


Figure 2: The MIRAGE-BENCH evaluation flowchart consisting of three steps: (i) Heuristic-based features on different dimensions for the baseline model response. (ii) Pairwise comparisons involving GPT-40 as a judge on a few queries to train a learning to rank model as a surrogate judge. (iii) After training, we utilize the learning to rank model to output the baseline model rankings on the complete set of queries in a synthetic arena-based leaderboard.

2023a; Chen et al., 2024c) or (ii) arena-based: where systems compete each other in a tournament and an LLM-based teacher is used as the judge (Rackauckas et al., 2024; Pradeep et al., 2024). Heuristic-based benchmarks are computationally cheaper to evaluate but require human preferences as gold truth for reference. They also face challenges in aggregating different metrics into a ranking order for models. On the other hand, arena-based benchmarks require a high performance LLM as a teacher (Zheng et al., 2023), which makes direct pairwise comparisons expensive for a large set of models. For example, using GPT-40 (OpenAI, 2023) to exhaustively evaluate a single query on 19 models requires  $\binom{19}{2} = 171$ comparisons and costs between \$5 and \$10 USD.

In our work, we get the best of both worlds by training a surrogate judge, a learning to rank model, e.g. random forest (Ho, 1995), using heuristic features to estimate an arena-based leaderboard obtained with a Bradley-Terry model (Hunter, 2004) from pairwise evaluations using an LLM as a judge (Zheng et al., 2023). We use boostrapping to obtain confidence bounds for better statistical estimates. After training, the learning to rank model can be used to estimate the performance of newer released models relaibly in the future without the expensive LLM as a judge. It also provides better interpretability and is easily retrainable with a different or newer set of heuristic features.

We develop MIRAGE-BENCH, a RAG benchmark across 18 languages for multilingual generation evaluation on Wikipedia. MIRAGE-BENCH is constructed from MIRACL (Zhang et al., 2023), a retrieval dataset containing human-generated queries and human-labeled relevance judgments

for Wikipedia passages. We benchmark 19 of the latest multilingual LLMs for use in multilingual RAG settings. Our evaluation flowchart adopted in MIRAGE-BENCH is listed in Figure 2. We evaluate seven heuristic features: (i) language detection, (ii) citation quality, (iii) support, (iv) reranker score, (v) answer overlap (traditional), (vi) answer overlap (LLM-measured), and (vii) fluency (LLMmeasured). Next, we use GPT-40 as a Judge (OpenAI, 2023) to evaluate our head-on pairwise RAG comparisons on a sampled set of a 100 queries for all languages. After this, we train a learning to rank model using random forest for each language, using the heuristic features as input and learn to output the Bradley-Terry logits as output. After training, we generate a "synthetic" arena-based leaderboard using the random forest model as a surrogate judge.

Our experimental results show that: (i) learning-to-rank model correlates well with GPT-40 arena-based leaderboards, achieving an average Kendall-Tau  $(\tau)$  score of 0.909. (ii) Proprietary and large (> 70B parameters) open-source models dominate the generation task by achieving the top-ranks in the MIRAGE-BENCH leaderboard. (iii) MIRAGE-BENCH training data, synthetically constructed using strong teachers such as GPT-40, is beneficial to improve smaller (7 or 8B parameters) open-source models on MIRAGE-BENCH.

The main contributions of our work are (i) building MIRAGE-BENCH, a benchmark to foster research towards multilingual RAG, incorporating a heuristic-based evaluation with an arena-based leaderboard using a trainable learning to rank model as a surrogate judge, and (ii) benchmarking 19 diverse multilingual-focused LLMs for the generation task in multilingual RAG.

	ar	bn	de	en	es	fa	fi	fr	hi	id	ja	ko	ru	sw	te	th	yo	zh
					Mir	AGE-BE	NCH Ev	aluation	Datase	t								
# Queries	1501	411	304	787	617	632	1183	343	350	939	797	213	1247	481	150	730	119	391
# Avg. Query Tokens	10.2	17.6	10.4	9.1	11.4	14.1	12.2	10.4	17.3	10.1	14.6	14.2	14.3	12.0	16.1	19.6	13.0	8.0
Avg. Rel. Passages / Q	2.0	2.1	2.6	2.8	4.3	2.1	2.0	2.1	2.1	3.1	2.2	2.6	2.8	1.9	1.2	1.8	1.2	2.5
Avg. Non Rel. Passages / Q	8.1	8.0	7.7	7.7	5.6	8.3	8.1	7.9	7.8	7.0	8.2	11.8	7.6	8.7	4.9	8.5	8.8	7.5
					Mı	RAGE-B	ENCH T	raining	Dataset									
# Queries	3468	1624	_	2857	2159	2104	2878	1137	1165	4054	3466	859	4567	1866	3283	2965	_	1311
# Avg. Query Tokens	10.2	17.4	_	8.7	11.2	14.2	11.8	10.4	17.1	9.9	14.5	14.9	14.3	12.0	16.1	19.3	_	7.9
Avg. Rel. Passages / Q	1.8	2.3	_	2.5	3.6	2.0	1.7	2.0	2.0	2.9	1.9	2.1	2.0	1.4	1.2	1.6	_	2.3
Avg. Non Rel. Passages / Q	5.5	7.9	_	7.5	5.3	8.3	5.3	8.0	7.9	7.1	7.9	12.4	5.2	3.5	4.3	5.6	_	7.6
# Avg. GPT-4o Context Tokens	105.5	144.5	_	137.9	203.4	133.8	129.4	180.1	149.4	100.8	140.2	136.4	90.0	106.4	54.6	86.0		137.9
# Avg. GPT-4o Answer Tokens	35.7	22.9	_	20.6	55.3	51.6	30.3	56.0	26.6	22.6	24.1	23.9	16.1	18.4	11.4	20.4	_	40.6

Table 1: Dataset statistics in MIRAGE-BENCH; All tokens are calculated using the GPT-40 tokenizer (OpenAI, 2023); (Rel.) denotes relevancy; (# Avg GPT-40) counts the tokens in the GPT-40 generated context and answer.

## 2 Related Work

Prior work on RAG evaluation has been conducted exclusively in English. For example, benchmarks such as ALCE (Gao et al., 2023a), FreshLLM (Vu et al., 2024), ClapNQ (Rosenthal et al., 2024), HAGRID (Kamalloo et al., 2023) and CRAG (Yang et al., 2024b), all include long-form answers for English-only queries and are based on collections containing documents from either the English Wikipedia, MS MARCO (Bajaj et al., 2016) or NQ (Kwiatkowski et al., 2019). Similarly, TREC 2024 RAG, an ongoing TREC competition for RAG evaluation is focused on English.<sup>2</sup>

Multilingual RAG. On the multilingual side, RAG has not been well studied in prior literature. The RGB benchmark (Chen et al., 2024c), is limited in language scope as it covers only one additional language: Chinese (zh). NeuCLIR (Mayfield et al., 2024) evaluates long-form report generation from participants in the upcoming 2024 track; but is limited to three languages. A concurrent work is BERGEN (Chirkova et al., 2024), which evaluates multilingual open-domain QA settings across 13 languages. In contrast, MIRAGE-BENCH evaluates the generation task<sup>3</sup> in the multilingual RAG pipeline on 18 languages, provides multilingual instruction-tuned data for RAG fine-tuning, and evaluates on high-quality queries in MIRACL.

**Learning to rank.** It is a supervised learning technique (Liu, 2010), where models are trained to provide an ordering between items in each list. The goal of constructing the ranking model is to rank new, unseen lists in a similar way to rankings in the training data (Turnbull, 2017). Models are trained

in either a pointwise, pairwise or listwise objective (Cao et al., 2007). In our work, we experiment with simpler approaches such as random forest as a surrogate judge to successfully learn to Bradley-Terry model coefficient produced by LLM as a judge. We keep experimentation with complex approaches such as LambdaMART (Burges, 2010) as a surrogate judge for future work.

## 3 MIRAGE-BENCH: A Multilingual RAG Benchmark

We select 18 languages in MIRAGE-BENCH as the starting point, representing an appropriate cross-section of the diversity of the languages spoken worldwide at this point. MIRAGE-BENCH serves as a comprehensive multilingual RAG benchmark focusing on the generation task evaluation. As shown in Table 1, MIRAGE-BENCH includes 11,195 evaluation pairs and 39,763 training pairs across 18 languages. We discuss the MIRAGE-BENCH evaluation and training datasets and highlight differences from the MIRACL dataset below:

#### 3.1 MIRAGE-BENCH Evaluation Dataset

MIRACL introduced in Zhang et al. (2023), is a monolingual retrieval dataset, i.e., queries and passages are both in the same language for passage retrieval. Queries are high-quality and generated by native speakers (Zhang et al., 2023). The annotation procedure in MIRACL is identical to TyDI-QA (Clark et al., 2020). The passage collection is constructed from language-specific Wikipedia corpora and parsed using WikiExtractor. The MIRAGE-BENCH evaluation dataset is constructed re-using the queries and oracle-judged passages available in the MIRACL development split.<sup>4</sup>

<sup>&</sup>lt;sup>2</sup>TREC 2024 RAG track: https://trec-rag.github.io/

<sup>&</sup>lt;sup>3</sup>As MIRAGE-BENCH is built from MIRACL, in our work we focus on the generation task in RAG using oracle retrieved passages, inspired by the "AG track" in the TREC 2024 RAG competition (https://trec-rag.github.io/).

<sup>&</sup>lt;sup>4</sup>We did not utilize the test split in MIRACL as the relevance judgments are not publicly available.

We incorporate two changes: (i) In Arabic (ar), we randomly sample a smaller subset of 1,501 out of 2,896 queries for uniformity in the number of pairs available across other languages. (ii) We filter out queries with zero non-relevant passages, i.e., we always include non-relevant passages, i.e., hard negatives, from MIRACL to make the MIRAGE-BENCH evaluation task challenging.<sup>5</sup>

### 3.2 MIRAGE-BENCH Training Dataset

The MIRAGE-BENCH training dataset is developed from the MIRACL (Zhang et al., 2023) training dataset. MIRACL only contains information on queries and relevant and non-relevant passages. In MIRAGE-BENCH, we reuse all the MIRACL training pairs available in 16 languages, except German (de) and Yoruba (yo). We convert the training dataset using a simple recipe into a multilingual instruction-tuned RAG dataset (Zhang et al., 2024; Niu et al., 2024). In our recipe, we filter the relevant passages, and keep them along with the input query to generate a zero-shot RAG output using strong teachers available including GPT-40 (OpenAI, 2023), Llama 3 (70B) (Dubey et al., 2024) and Mixtral (8x22B) (Jiang et al., 2024). After generation, we include non-relevant passages within our prompt as "distracting and noisy information", to help improve the quality of the MIRAGE-BENCH training dataset. Note that, since we convert a retrieval dataset, we do not have human-annotated answers for questions in MIRAGE-BENCH.

#### 3.3 Distinction and Extension from MIRACL

MIRACL introduced in Zhang et al. (2023) is a monolingual retrieval dataset, which evaluates the "retrieval task", i.e. given a user query and a Wikipedia corpus, MIRACL contains human-annotated relevance judgements to evaluate for query-passage level relevancy using retrieval models, e.g., sparse models like BM25 (Robertson and Zaragoza, 2009) or bi-encoders like mDPR (Karpukhin et al., 2020), or late-interaction models like ColBERT (Khattab and Zaharia, 2020).

In contrast, MIRAGE-BENCH evaluates the "generation task" in RAG, requiring LLMs to generate a summarized answer given the query and context available from retrieved passages. In our work, we re-use the queries and oracle relevance-judgments from MIRACL and focus solely on evaluating the

generation task in RAG measuring answers using both heuristics and LLM as a judge.

## 4 Multilingual RAG Evaluation

#### 4.1 Heuristic-based Evaluation

Multilingual generation in RAG requires the evaluation of multiple dimensions. For example, whether a system's response provides the correct final answer or cites the relevant documents, a single metric alone is *not sufficient* to capture the comprehensive evaluation required for RAG systems. Inspired by other recent works (Kiela et al., 2021; Santhanam et al., 2023; Gao et al., 2023a), we introduce five *deterministic* features and two *LLM* as a judge features for evaluation. We additionally provide details about each feature in Appendix A.

Language detection. We compute the probability of a system's response in the required target language with langid (Lui and Baldwin, 2012). We compute two metrics: language detection (target language) and English detection.

Citation quality. Using passage-level relevance judgments for all queries (or qrels) information available in MIRACL, we evaluate whether the system's response cites the relevant passages, crucial for measuring faithfulness. We compute and evaluate: Recall@k and MAP@k, where k = 10.

**Support.** Grounding is necessary to avoid hallucinations in the system's response. Support evaluation (Gao et al., 2023a) checks whether each sentence is supported by cited passages using a multilingual NLI model.<sup>6</sup> We compute the probability of the *entailment* and *neutral* score, macro-averaged across the sentence-citation pairs.

**Reranker score.** The reranker score measures the average similarity (can be greater than 1.0) between the query and the passages cited within the system's response. We compute the reranker score using a multilingual reranker model,<sup>7</sup> macro-averaged across the query-passage pairs.

Answer overlap. Having the correct answer is crucial in the RAG system's response. Since MIRAGE-BENCH does not include a human-labeled answer, we use the generated answer from GPT-4 (OpenAI, 2023) as the gold truth. We compute two traditional answer overlap metrics: SacreBLEU

<sup>&</sup>lt;sup>5</sup>An exception here is Telugu (te), where only 78 queries have at least a single non-relevant passage, we sample and include 72 additional queries with only relevant passages.

<sup>&</sup>lt;sup>6</sup>Fine-tuned mDeBERTa-v3-base XNLI model (He et al., 2023): MoritzLaurer/mDeBERTa-v3-base-xnli-multilingual-nli-2mil7

<sup>&</sup>lt;sup>7</sup>Reranker (Chen et al., 2024b): BAAI/bge-reranker-v2-m3

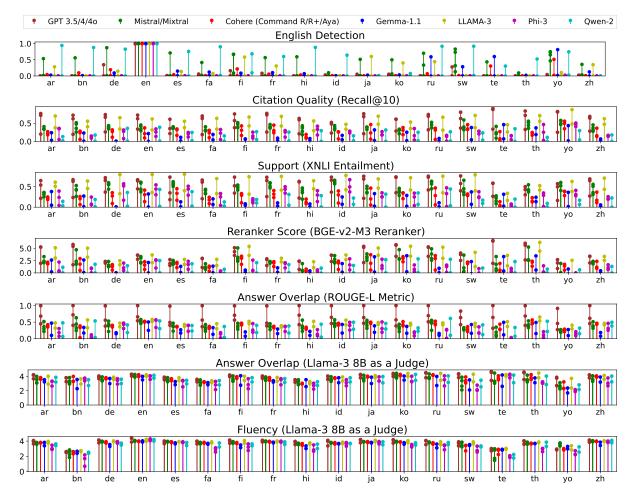


Figure 3: Lollipop plots denoting the average heuristic-based feature scores achieved by baselines in MIRAGE-BENCH. *x*-axis denotes the languages in MIRAGE-BENCH; whereas *y*-axis plots every heuristic feature value. models in the same family are represented as the same color in a lollipop (as multiple circles). Figure 9 provides lollipop plots for all eleven heuristic-based features.

(Papineni et al., 2002) and ROUGE-L (Lin, 2004) to measure the lexical word overlap between the gold truth answer (here, GPT-4 answer) and the system's response.

Answer overlap (LLM-measured). In addition, we evaluate using Llama-3 (8B) (Dubey et al., 2024), an open-source LLM as a judge evaluator in a pointwise setup, providing a semantic word overlap integer score in the range [1, 5]. The answer overlap prompt description is listed in Figure 11.

Fluency (LLM-measured). It measures for grammatical correctness and idiomatic word choices in the system's response. As previously mentioned, we use Llama-3 (8B) (Dubey et al., 2024) in a pointwise setup, proving an integer score in [1, 5]. The fluency prompt description is listed in Figure 12.

#### 4.2 Arena-based Evaluation

Heuristic-based evaluation metrics often rely on a gold standard for evaluation. Tasks such as text

retrieval (Bajaj et al., 2016; Thakur et al., 2021) require human-labeled relevance judgments, and similarly, NLP tasks such as machine translation (Stahlberg, 2020), require human-annotated translations. As human preferences are seldom available in numerous applications, using LLMs as a judge (Zheng et al., 2023; Chen et al., 2024a; Chiang et al., 2024) is becoming a defacto approach for arena-based evaluation of models.

Pairwise LLM as a judge. Following prior works on arena-based evaluation in RAG (Rackauckas et al., 2024; Pradeep et al., 2024), we evaluate two system's responses in a head-on battle by computing pairwise judgments with LLM as a judge. We reuse the prompt template from RAGElo (Rackauckas et al., 2024) with minor additions. LLM as a judge evaluator includes three types of biases (Zheng et al., 2023): (i) verbosity bias (Wu and Aji, 2023) (ii) self-enhancement bias (Xu et al., 2024; Panickssery et al., 2024) and (iii) position bias

(Wang et al., 2024). We successfully avoid the verbosity bias, as the RAG evaluation has fixed evaluation criteria requiring sentence-level citations and answers (Pradeep et al., 2024) and the position bias by randomly swapping the position of two models. Computing exhaustive pairwise comparisons with GPT-40 (OpenAI, 2023) as a teacher is expensive, we keep it as future work to include better and more teachers to eliminate the self-enhancement bias.

### 4.3 Learning to Approximate Rankings

There is no predefined way to aggregate the heuristic features to provide an overall leaderboard ranking in MIRAGE-BENCH. Averaging the scores is too simplistic, as features measure different aspects of RAG evaluation. On the other hand, arenabased evaluations provide ranked leaderboards but are computationally expensive to compute with a strong teacher model. To avoid computational costs, smaller models as teachers have been proposed (Thakur et al., 2024a; Ni et al., 2024). Motivated by similar observations, we train a learning to rank model as a surrogate judge to approximate the Bradley-Terry model coefficients (Hunter, 2004) learned from an arena-based evaluation that uses GPT-40 for pairwise judgments.

**Regression model.** While the heuristic RAG features introduced in Section 4.1 can be computed efficiently and without the reliance on proprietary models, inducing a ranking from pairwise comparisons via a Bradley-Terry model is computationally expensive and requires access to a high-performance LLM. Furthermore, in Section 6.3, we demonstrate that the ranking accuracy, measured by the Kendall-Tau  $(\tau)$  coefficient, degrades rapidly when subsampling tournament matches. Therefore we investigate whether a regression model can be successfully trained to approximate the Bradley-Terry logits using heuristic features.

The procedure, detailed in Algorithm 1, simulates  $N_t$  tournaments, each involving a total of  $N_l$  models and  $N_q$  queries. For each query, judgments are obtained for all  $\binom{N_l}{2}$  pairings of models. We employ bootstrapping on the query selection process to estimate the variance in the  $R^2$  metric in the regression models' approximations of the Bradley-Terry coefficients over a randomly-sampled holdout set, LLM $_{predict}$ .

We randomly selected two models, Gemma 1.1 (2B) as Llama-3 (70B) as holdout models. For English, we observed an average  $\bar{R}^2 = 0.971$  with

Algorithm 1 Simulate Tournaments and Fit Models

```
 \begin{array}{ll} \text{1: } \textbf{for } i \in [N_t] \, \textbf{do} \\ \text{2: } & M_{BT}^i \leftarrow \text{TOURNAMENT}(N_q) \\ \text{3: } & X_t, Y_t \leftarrow \text{DATASET}(\text{LLM}_{train}, M_{BT}^i) \\ \text{4: } & X_p, Y_p \leftarrow \text{DATASET}(\text{LLM}_{predict}, M_{BT}^i) \\ \text{5: } & M_{reg}^i \leftarrow \text{FIT}(X_t, Y_t) \\ \text{6: } & R_2^i \leftarrow M_{reg}^i(X_p, Y_p) \\ \text{7: } & \textbf{end for} \\ \text{8: } & M_{BT} \leftarrow [M_{BT}^1; M_{BT}^2; ...; M_{BT}^{N_t}] \\ \text{9: } & M_{reg} \leftarrow [M_{reg}^1; M_{reg}^2; ...; M_{reg}^{N_t}] \\ \text{10: } & R_2 \leftarrow [R_2^1; R_2^2; ...; R_2^{N_t}] \\ \end{array}
```

**Note:** Refer to Section 4.3 for a definition of each of the variables. The TOURNAMENT function runs a battle arena, sampling q queries, and returning the learned Bradley-Terry model. The DATASET function accepts a set of LLMs and a learned Bradley-Terry model. It returns X, the heuristic RAG feature values, and Y, the Bradley-Terry coefficients for each LLM model. After simulating  $N_t$  tournaments, the array  $R_2$  contains the  $R^2$  errors for each of the  $N_t$  regression models.

a 95% confidence interval of [0.905, 0.999], while for Bengali, we observed  $\bar{R}^2=0.937$  with a 95% confidence interval of [0.766, 0.998]. All scores with 95% confidence intervals are listed in Table 5. Taken together, these results indicate that the training procedure is fairly robust with  $N_q=100$ .

#### 5 Experimental Settings

#### **5.1** Multilingual Baselines

Existing state-of-the-art LLMs are either Englishonly or support a limited set of languages, predominantly due to the *curse of multilinguality* for large models (Conneau et al., 2020), i.e., it is unclear how well-existing LLMs perform on RAG across a wide variety of languages, due to scarce availability of multilingual instruction tuning datasets. We experiment with models from seven different families, containing proprietary and open-sourced models. Wherever possible, we benchmark the *instruction-tuned* version if available. Please refer to Appendix B for more details on baselines.

- **OpenAI:** We evaluate the GPT-3.5-turbo, GPT-4, and GPT-40 models (OpenAI, 2023) using Azure OpenAI service.
- **Mistral:** We evaluate the Mistral-Instruct-v0.2 and v0.3 (7B) (Jiang et al., 2023), Mixtral (8x7B) and Mixtral (8x22B) (Jiang et al., 2024).
- Cohere: We evaluate the Command-R (35B), Command-R+ (104B) and Aya-23 (35B) models (Aryabumi et al., 2024).

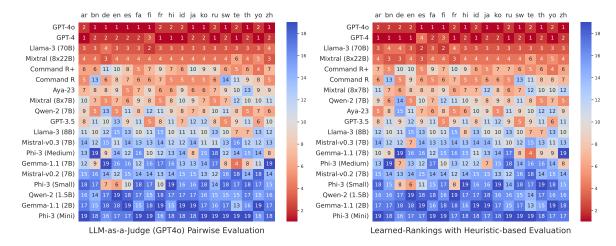


Figure 4: MIRAGE-BENCH arena-based leaderboards: (left) Bradley-Terry model coefficients with rankings with GPT-40 as a judge pairwise judgments on a subset of 100 sampled queries; (right) Synthetic rankings using heuristic-based features and a learning to rank model. Each highlighted value is the rank of the model on the language (the lower the better). Models have been sorted by lowest to highest average rank across all 18 languages.

- **Gemma:** We evaluate the Gemma 1.1 instruct (2B) and (7B) models (Mesnard et al., 2024).
- Llama-3: We evaluate the Llama-3 instruct (8B) and (70B) models (Dubey et al., 2024).
- **Phi-3:** We evaluate all models from the Phi 3 instruct series: Medium (14B), Small (7B), and Mini (3.8B) (Abdin et al., 2024).
- **Qwen-2:** We evaluate two versions of the Qwen-2 model: 1.5B and 7B (Yang et al., 2024a).

#### **5.2** Evaluation Details

**Prompt template.** We internally optimized<sup>8</sup> the ChatQA prompt template (Liu et al., 2024), to include in-text citations of the context passages following the IEEE format (Kamalloo et al., 2023). In MIRAGE-BENCH, we have about 10 passages annotated in the oracle setting. Therefore, we trim each passage available and take the first 800 tokens to fit all passages within a fixed context length of 8192 tokens. following prior work in Shi et al. (2023), the prompt requires the LLM to explain the multilingual generation task starting with "##Reason" and the answer itself starting with "##Answer". Utilizing this output format has its advantages in easily parsing the generated answer and the rationale behind the answer. The prompt template is shown in Figure 10. For GPT-40 as a judge pairwise evaluation, we modified the prompt template available in RAGEval (Rackauckas et al., 2024). The prompt template is shown in Figure 13.

Lang.	$\tau$	Lang.	τ	Lang.	τ	Lang.	τ	Lang.	τ
ar	0.951	bn	0.874	de	0.825	en	0.835	es	0.876
fa	0.924	fi	0.949	fr	0.914	hi	0.946	id	0.896
ja	0.892	ko	0.950	ru	0.849	sw	0.958	te	0.938
		th	0.946	yo	0.906	zh	0.941		
			Avg.	Kendall '	Tau (τ) =	0.909			

Table 2: Kendall  $\tau$  rank correlation between GPT-40 as a judge pairwise Bradley-Terry model and arenabased ranking leaderboard using heuristics and trained regression model for all languages in MIRAGE-BENCH.

## 6 Experimental Results

#### 6.1 Heuristic-based Results

Figure 3 shows lollipop plots indicating the average heuristic-feature value (y-axis) distribution across all languages (x-axis). In English detection, smaller LLMs such as Gemma-1.1 (2B) do not generate output in the required target language, but rather rely on English. Next, for citation quality and support evaluation, OpenAI models achieve better Recall@10 and Entailment scores (except Llama-3 (70B) for a few languages), indicating baseline responses include grounded citations from relevant passages. In contrast, models such as Qwen-2 or Gemma-1.1 tend to under-cite in their response. Similar trends are observed for the reranker score.

Furthermore, we observe OpenAI models achieve a higher word overlap in the ROUGE-L metric (GPT-4 used as ground truth) and for Llama-3 (8B) as a judge, we observe rather less variance in scores across models. In Fluency, we observe a majority of the baselines are rather fluent, except Bengali (bn), Telugu (te), and Yoruba (yo).

<sup>&</sup>lt;sup>8</sup>A majority of the prompt optimization was internal and based on eye-balling RAG responses across LLMs.

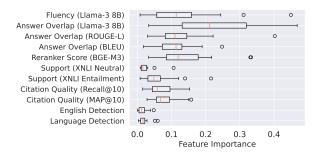


Figure 5: Boxplot with the feature importance value (averaged across 18 languages in MIRAGE-BENCH) observed by the learning to rank (random forest) model.

#### 6.2 Arena-based Results

Figure 4 (left) shows the arena-based leaderboard using bootstrapping and Bradley-Terry modeling after conducting 200 tournaments and sampling 100 matches per tournament on a subset of 100 queries using GPT-40 pairwise comparisons. We observe that proprietary models such as GPT-40 and GPT-4, and larger models such as LLAMA-3 (70B) and Mixtral (8x22B), are slightly better than other baselines on MIRAGE-BENCH. Baseline rankings across languages are usually stable; with a few notable exceptions such as Gemma-1.1 (7B) which achieves a rank of 4 in Telugu. Command R (35B) performs poorly in low-resource languages such as Bengali (rank 13) or Swahili (rank 14). The complete bradley-terry model coefficient logits and 95% confidence intervals across all 18 languages in MIRAGE-BENCH are provided in Table 7 and Table 8 in the Appendix.

Synthetic rankings using the learning to rank model. Figure 4 (right) plots the overall synthetic average rankings on all queries using heuristic-based features trained with a random forest learning to rank model in MIRAGE-BENCH. The learned-ranking leaderboard highly correlates to the GPT-40 as a judge leaderboard, achieving an average Kendall-Tau ( $\tau$ ) rank correlation = **0.909**, by training on 17 models during training and 2 models as holdout for every language. Individual language-specific Kendall-Tau rank correlation scores are listed in Table 2. This provides evidence of the efficacy in training a random forest model as a surrogate judge on heuristic features with bootstrapping to approximately learn the Bradley-Terry logits.

**Heuristic feature importance.** In Figure 5, we plot the average feature importance achieved by our Random Forest regression model across all 18 languages. We observe using Llama-3 (8B) as a

Model / Language	ar	bn	fi	ja	ko	ru	te
Trai	n $\mathbb{R}^2$ on	random	ly selecte	d fifteer	n models		
Random Forest	0.97	0.96	0.97	0.96	0.97	0.95	0.97
Linear Regression	0.98	0.98	0.98	1.00	0.97	0.99	1.00
MLP Regressor	0.97	0.98	0.97	0.96	0.96	0.98	0.99
XGB Regressor	1.00	1.00	1.00	1.00	1.00	1.00	1.00
SVR	0.75	0.77	0.81	0.59	0.67	0.73	0.39
Holdout	$R^2$ on for	our rand	omly sele	ected he	ld out mo	dels	
Random Forest	0.50	0.41	0.49	-0.03	0.45	0.07	0.83
Linear Regression	-1.92	-5.45	-19.38	-0.06	-26.53	-2.13	0.31
MLP Regressor	0.33	0.37	0.45	-0.76	-0.04	-0.48	0.78
XGB Regressor	-0.02	0.44	0.22	-1.33	-0.09	-0.80	0.59
SVR	-0.03	0.48	0.64	-0.53	-0.09	0.15	-0.10

Table 3: Train and Holdout  $\mathbb{R}^2$  scores using different regression models. Each experiment has been repeated 50 times with four held-out models.

judge, for fluency and answer overlap on average are most important heuristic features. Similarly, deterministic answer-overlap and reranking-based metrics are equally important. Some heuristic features such as language detection, and neutral score in support evaluation obtain the least importance. We potentially observe all answer overlap heuristic features achieve a high score indicating that the "answer" portion in the system's RAG response is crucial for evaluation and is highly correlated with GPT-40 as a judge for pairwise evaluations.

#### 6.3 Ablations & Discussion

To better understand the gaps observed during training of the regression model, we conduct further ablations on a subset of seven languages including Arabic (ar), Bengali (bn), Finnish (fi), Japanese (ja), Korean (ko), Russian (ru) and Telugu (te):

Regression model choice. We compare several learning to rank models as choices for learning the Bradley-Terry model coefficients. We conduct our experiments on the train set, where models contain pairwise judgments, and on a randomly sampled holdout set, a realistic scenario, with no available training data. We evaluate Random Forest, Linear Regression, MLP Regressor, XGB Regressor, and SVR. All learning to rank models are implemented via scikit-learn.9 The results are depicted in Table 3. Random forest achieves the best  $R^2$  value on the holdout set for 4/7 languages. SVR also achieves a  $R^2$  value on the holdout subset, however, underperforms random forest on the training dataset. Other baselines, such as XGB Regressor and MLP Regressor show signs of overfitting on the training subset, thereby underperforming random forest on the holdout subset in MIRAGE-BENCH.

<sup>&</sup>lt;sup>9</sup>https://scikit-learn.org/stable/supervised\_learning.html

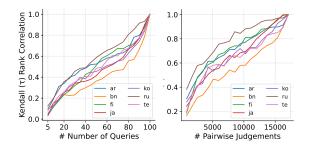


Figure 6: Sampling experiments to reduce computation cost. (left) reduces the number of queries whereas (right) reduces the pairwise judgments.

Features / Language	#F	ar	bn	fi	ja	ko	ru	te
(All) Features	11	0.938	0.867	0.921	0.881	0.921	0.826	0.918
(W/o) LLM as a Judge	9	0.912	0.866	0.898	0.853	0.891	0.811	0.904
(W/o) Low. Correlation	7	0.951	0.867	0.923	0.885	0.929	0.829	0.940
(Only) LLM as a Judge	2	0.948	0.728	0.907	0.884	0.916	0.851	0.872

Table 4: Kendall Tau  $(\tau)$  scores using different features for training the random forest regression model.

Non-exhaustive pairwise comparisons lead to performance degradation. Exhaustive pairwise comparisons across a subset of 19 models in MI-RAGE-BENCH using GPT-40 as a teacher for "all" queries is quite expensive. To avoid this, we investigate whether really (all) pairwise exhaustive comparisons are necessary. We utilize two sampling techniques: (i) full pairwise judgments on a subsample of 100 queries, e.g., 20 or 50 queries; (ii) partially judge a non-exhaustive random sample of the pairwise judgments across 100 queries, e.g., only 50% of all the exhaustive pairwise combinations. Both results are shown in Figure 6. we observe that Kendall-Tau  $(\tau)$  correlations increase linearly with queries and pairwise judgments. In summary, an exhaustive pairwise comparison and a sufficient amount of queries, such as 100, is necessary without impacting the leaderboard rankings.

Are all heuristic features necessary? We experiment with the set of features used for learning to rank model training as a surrogate judge. We evaluate four training configurations: (i) all features (ii) without LLM-measured features (iii) without language detection and support, i.e., the low-correlation features observed in Figure 5, and (iv) including only LLM-measured features. From Table 4, we observe that removing low-correlated heuristic RAG features, actually helps the learning to rank model to learn better leading to a conclusion that not necessarily all features are important. Next, removing the LLM-measured features completely or only keeping them decreases the Kendall-Tau  $(\tau)$  correlation score in MIRAGE-BENCH.

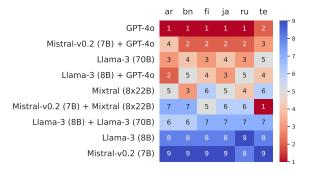


Figure 7: Approximate rankings using heuristic features after fine-tuning Llama-3 (8B) and Mistral-v0.2 (7B) on MIRAGE-BENCH dataset across four configurations.

**Does fine-tuning on MIRAGE-BENCH training data help?** We evaluate three variants of the MIRAGE-BENCH training dataset using two backbones: Mistral-v0.2 (7B) and Llama-3 (8B). We fine-tune the MIRAGE-BENCH training datasets using (i) both on GPT-4o, (ii) Llama-3 (8B) on Llama-3 (70B), and (iii) Mistral-v0.2 (7B) on Mixtral (8x22B). From Figure 7, we observe that GPT-4o is a strong teacher, Mistral-v0.2 (7B) fine-tuned on GPT-4o distilled training data achieves the rank 2 outperforming the Llama-3 (70B) model. This shows MIRAGE-BENCH training data is useful to improve the generation task in MIRAGE-BENCH.

## 7 Conclusion

We present MIRAGE-BENCH, a multilingual RAG benchmark for 18 languages aimed towards accelerating research on multilingual RAG. In MIRAGE-BENCH, we evaluate the multilingual generation part within RAG and aggregate traditional heuristic-based features to train a lightweight learning to rank model as a surrogate judge to learn a Bradley Terry model with GPT-40 pairwise judgments. Our results indicate a strong correlation between our surrogate judge and original LLM as a judge arenabased leaderboard demonstrating the effectiveness of our random forest model trained using RAG-based heuristic features to emulate a much expensive LLM as a judge pariwse rankings.

Using our arena-based leaderboard, we observed a majority of proprietary and open-sourced larger models currently dominate on the MIRAGE-BENCH benchmark. Instruction tuning on MIRAGE-BENCH training data helps improve the performance of open-sourced and smaller models on MIRAGE-BENCH. Instruction-tuned Mistral-v0.2 (7B) is able to outperform the Llama 3 (70B) baseline on MIRAGE-BENCH.

#### 8 Limitations

MIRAGE-BENCH is one of the first holistic multilingual RAG benchmarks. Although not perfect, we below discuss a set of limitations in our work:

- In MIRAGE-BENCH, we focused on benchmarking the generation task in RAG, we did not evaluate the multilingual retrieval task and its error propagating on the multilingual generation task.
- In the future, we wish to evaluate diverse LLMs as teachers such as Claude-3.5 (sonnet) (Anthropic, 2024) or Gemini Pro (Anil et al., 2023), currently we only included a single teacher, GPT-40, as a judge for evaluation. This can cause self-enhancement bias in our experiments towards GPT-40 and similar models in the family (GPT-4 and GPT-3.5).
- In our heuristic evaluation, we only considered a small subset of features. We did not explore more recent RAG evaluation techniques such as nugget-based evaluation (Lin and Demner-Fushman, 2005).
- Lastly, MIRAGE-BENCH dataset does not provide human-labeled answers for queries across all languages and is limited to Wikipedia, unlike recent RAG benchmarks which use the human answer for RAG-based evaluation or cover a wider variety of domains in English.

## Acknowledgements

This research is supported by the Canada First Research Excellence Fund and the Natural Sciences and Engineering Research Council (NSERC) of Canada. We thank Ka Wong for helping with bootstrapping during the regression model's training.

## References

Marah I Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat S. Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Caio César Teodoro Mendes, Weizhu Chen, Vishrav Chaudhary, Parul Chopra, Allie Del Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Dan Iter, Amit Garg, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Chen Liang, Weishung Liu, Eric Lin, Zeqi Lin, Piyush Madan, Arindam Mitra, Hardik

Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Xia Song, Masahiro Tanaka, Xin Wang, Rachel Ward, Guanhua Wang, Philipp Witte, Michael Wyatt, Can Xu, Jiahang Xu, Sonali Yadav, Fan Yang, Ziyi Yang, Donghan Yu, Chengruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *CoRR*, abs/2404.14219.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. 2023. Gemini: A family of highly capable multimodal models. CoRR, abs/2312.11805.

Anthropic. 2024. Claude 3.5 sonnet.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, Kelly Marchisio, Max Bartolo, Sebastian Ruder, Acyr Locatelli, Julia Kreutzer, Nick Frosst, Aidan N. Gomez, Phil Blunsom, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2024. Aya 23: Open weight releases to further multilingual progress. *CoRR*, abs/2405.15032.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2016. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. 2022. Improving language models by retrieving from

- trillions of tokens. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR.
- Chris J.C. Burges. 2010. From RankNet to LambdaRank to LambdaMART: An overview. Technical Report MSR-TR-2010-82.
- Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, page 129–136, New York, NY, USA. Association for Computing Machinery.
- Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024a. Humans or LLMs as the judge? A study on judgement biases. *CoRR*, abs/2402.10669.
- Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024b. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 2318–2335, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.
- Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024c. Benchmarking large language models in retrieval-augmented generation. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 17754–17762. AAAI Press.
- Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael I. Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: An open platform for evaluating LLMs by human preference. In Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024. OpenReview.net.
- Nadezhda Chirkova, David Rau, Hervé Déjean, Thibault Formal, Stéphane Clinchant, and Vassilina Nikoulina. 2024. Retrieval-augmented generation in multilingual settings. In *Proceedings of the 1st Workshop on Towards Knowledgeable Language Models (KnowLLM 2024)*, pages 177–188, Bangkok, Thailand. Association for Computational Linguistics.
- Elizabeth Clark, Shruti Rijhwani, Sebastian Gehrmann, Joshua Maynez, Roee Aharoni, Vitaly Nikolaev, Thibault Sellam, Aditya Siddhant, Dipanjan Das, and Ankur P. Parikh. 2023. SEAHORSE: A multilingual, multifaceted dataset for summarization evaluation. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP

- 2023, Singapore, December 6-10, 2023, pages 9397–9413. Association for Computational Linguistics.
- Jonathan H. Clark, Jennimaria Palomaki, Vitaly Nikolaev, Eunsol Choi, Dan Garrette, Michael Collins, and Tom Kwiatkowski. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Trans. Assoc. Com*put. Linguistics, 8:454–470.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8440–8451. Association for Computational Linguistics.
- Tri Dao. 2024. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024.* OpenReview.net.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham,

Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujiwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan,

Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen,

- Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models.
- Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: long form question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3558–3567. Association for Computational Linguistics.
- Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023a. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 6465–6488. Association for Computational Linguistics.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. 2023b. Retrieval-augmented generation for large language models: A survey. *CoRR*, abs/2312.10997.
- Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML* 2020, 13-18 July 2020, Virtual Event, volume 119 of Proceedings of Machine Learning Research, pages 3929–3938. PMLR.
- Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.
- Tin Kam Ho. 1995. Random decision forests. In *Third International Conference on Document Analysis and Recognition, ICDAR 1995, August 14 15, 1995, Montreal, Canada. Volume I*, pages 278–282. IEEE Computer Society.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR* 2022, Virtual Event, April 25-29, 2022. OpenReview.net.
- David R. Hunter. 2004. MM algorithms for generalized Bradley-Terry models. *The Annals of Statistics*, 32(1):384 406.
- Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 23, 2021*, pages 874–880. Association for Computational Linguistics.

- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.
- Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *CoRR*, abs/2401.04088.
- Ehsan Kamalloo, Aref Jafari, Xinyu Zhang, Nandan Thakur, and Jimmy Lin. 2023. HAGRID: A human-llm collaborative dataset for generative information-seeking with attribution. *CoRR*, abs/2307.16883.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for opendomain question answering. In *Proceedings of the* 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6769–6781, Online. Association for Computational Linguistics.
- Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. In 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net.
- Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 39–48, New York, NY, USA. Association for Computing Machinery.
- Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, pages 4110–4124. Association for Computational Linguistics.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti,

- Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7:452–466.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles, SOSP 2023, Koblenz, Germany, October 23-26, 2023*, pages 611–626. ACM.
- Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.
- Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Jimmy Lin and Dina Demner-Fushman. 2005. Automatically evaluating answers to definition questions. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 931–938, Vancouver, British Columbia, Canada. Association for Computational Linguistics.
- Tie-Yan Liu. 2010. Learning to rank for information retrieval. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*, page 904. ACM.
- Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.
- Zihan Liu, Wei Ping, Rajarshi Roy, Peng Xu, Chankyu Lee, Mohammad Shoeybi, and Bryan Catanzaro. 2024. ChatQA: Building GPT-4 level conversational QA models. *CoRR*, abs/2401.10225.
- Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the System Demonstrations, July 10, 2012, Jeju Island, Korea*, pages 25–30. The Association for Computer Linguistics.

- Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. PEFT: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft.
- James Mayfield, Eugene Yang, Dawn J. Lawrie, Sean MacAvaney, Paul McNamee, Douglas W. Oard, Luca Soldaini, Ian Soboroff, Orion Weller, Efsun Selin Kayi, Kate Sanders, Marc Mason, and Noah Hibbler. 2024. On the evaluation of machine-generated reports. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 1904–1915. ACM.
- Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Cristian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan. Jeremy Chen, Johan Ferret, Justin Chiu, and et al. 2024. Gemma: Open models based on gemini research and technology. CoRR, abs/2403.08295.
- Microsoft. 2023. Reinventing search with a new AIpowered microsoft bing and edge, your copilot for the web.
- Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. 2024. MixEval: Deriving wisdom of the crowd from LLM benchmark mixtures. *CoRR*, abs/2406.06565.
- Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878, Bangkok, Thailand. Association for Computational Linguistics.
- OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.
- Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. LLM evaluators recognize and favor their own generations. *CoRR*, abs/2404.13076.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the* 40th Annual Meeting of the Association for Computational Linguistics, pages 311–318, Philadelphia,

Pennsylvania, USA. Association for Computational Linguistics.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaïd Harchaoui. 2021. MAUVE: measuring the gap between neural text and human text using divergence frontiers. In Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pages 4816–4828.

Ronak Pradeep, Nandan Thakur, Sahel Sharifymoghaddam, Eric Zhang, Ryan Nguyen, Daniel Campos, Nick Craswell, and Jimmy Lin. 2024. Ragnarök: A reusable RAG framework and baselines for TREC 2024 retrieval-augmented generation track. *CoRR*, abs/2406.16828.

Zackary Rackauckas, Arthur Câmara, and Jakub Zavrel. 2024. Evaluating RAG-fusion with RAGElo: an automated elo-based framework. *CoRR*, abs/2406.14783.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Sara Rosenthal, Avirup Sil, Radu Florian, and Salim Roukos. 2024. CLAPNQ: cohesive long-form answers from passages in natural questions for RAG systems. *CoRR*, abs/2404.02103.

Keshav Santhanam, Jon Saad-Falcon, Martin Franz, Omar Khattab, Avi Sil, Radu Florian, Md Arafat Sultan, Salim Roukos, Matei Zaharia, and Christopher Potts. 2023. Moving beyond downstream task accuracy for information retrieval benchmarking. In Findings of the Association for Computational Linguistics: ACL 2023, pages 11613–11628, Toronto, Canada. Association for Computational Linguistics.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2023. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Felix Stahlberg. 2020. Neural machine translation: A review. *J. Artif. Intell. Res.*, 69:343–418.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill,

Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving open language models at a practical size.

Aman Singh Thakur, Kartik Choudhary, Venkat Srinik Ramayapally, Sankaran Vaidyanathan, and Dieuwke Hupkes. 2024a. Judging the judges: Evaluating alignment and vulnerabilities in LLMs-as-Judges. *CoRR*, abs/2406.12624.

Nandan Thakur, Luiz Bonifacio, Xinyu Zhang,

- Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Boxing Chen, Mehdi Rezagholizadeh, and Jimmy Lin. 2023. NoMIRACL: Knowing when you don't know for robust multilingual retrieval-augmented generation. *CoRR*, abs/2312.11361.
- Nandan Thakur, Jianmo Ni, Gustavo Hernandez Abrego, John Wieting, Jimmy Lin, and Daniel Cer. 2024b. Leveraging LLMs for synthesizing training data across many languages in multilingual dense retrieval. In Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 7699–7724, Mexico City, Mexico. Association for Computational Linguistics.
- Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks* 2021, December 2021, virtual.
- Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Shengyi Huang, Kashif Rasul, Alvaro Bartolome, Alexander M. Rush, and Thomas Wolf. 2023. The Alignment Handbook.
- Doug Turnbull. 2017. Learning to rank 101 linear models.
- Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry W. Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc V. Le, and Thang Luong. 2024. FreshLLMs: Refreshing large language models with search engine augmentation. In Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024, pages 13697–13720. Association for Computational Linguistics.
- Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. 2024. Large language models are not fair evaluators. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024, pages 9440–9450. Association for Computational Linguistics
- Minghao Wu and Alham Fikri Aji. 2023. Style over substance: Evaluation biases for large language models. *CoRR*, abs/2307.03025.
- Wenda Xu, Guanglei Zhu, Xuandong Zhao, Liangming Pan, Lei Li, and William Wang. 2024. Pride and prejudice: LLM amplifies self-bias in self-refinement. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15474–15492. Association for Computational Linguistics.

- An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024a. Qwen2 technical report.
- Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Ethan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, Lei Chen, Nicolas Scheffer, Yue Liu, Nirav Shah, Rakesh Wanga, Anuj Kumar, Wen tau Yih, and Xin Luna Dong. 2024b. CRAG comprehensive rag benchmark.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 November 4, 2018*, pages 2369–2380. Association for Computational Linguistics.
- Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E. Gonzalez. 2024. RAFT: Adapting language model to domain specific RAG. In *First Conference on Language Modeling*.
- Xinyu Zhang, Nandan Thakur, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Mehdi Rezagholizadeh, and Jimmy Lin. 2023. MIRACL: A Multilingual Retrieval Dataset Covering 18 Diverse Languages. Transactions of the Association for Computational Linguistics, 11:1114–1131.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and chatbot arena. In Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 16, 2023.

## A Heuristic-based Evaluation: Features & Additional Details

- 1. Language Identification: In a multilingual RAG system, the output response should ideally be in the same language in which the user asked their query. To capture this feature, we attempt to identify which natural language the output response is in. We use langid (Lui and Baldwin, 2012), an off-the-shelf language detecting Python library for detecting the language of the long-form RAG answer. We use the probability of the target language detected as the score for language identification, i.e.,  $\hat{p} = langid(a, t)$ , where t denotes the target language and t denotes the long-form answer.
- 2. Citation Quality: A multilingual RAG system must cite information from the relevant passages within their answers, to improve faithfulness and reduce hallucinations. We capture whether the passages (using relevance judgments provided in MIRACL (Zhang et al., 2023)) are cited in the multilingual generation task. For scoring, we compute the Recall@k and MAP@k score, where Recall@k is 1.0 for a generated answer a, if and only if a cites all available relevant passages, Similarly, the MAP@10 score measures the percentage of relevant passages within the top-k cited passages.
- 3. Support: RAG systems have been shown to hallucinate across retrieval-augmented generation tasks, especially when provided with non-relevant contexts (Thakur et al., 2023). Grounding is necessary to avoid hallucinations. We compute the grounding score of every sentence  $s_j$  in generated answer A along with the cited context  $c_j$  using the multilingual NLI model, which computes the similarity score as a probability of either *entailment*, *neutral* or *contradiction*. The entailment denotes the generated sentence in the long-form answer, which entails the cited passage within its response.
- **4. Reranker Score:** The reranker score measures the semantic similarity between the user query and the cited passages in the system's response. If the cited passages are relevant in answering the query, the reranker model would output a higher similarity score. We utilize a multilingual open-source reranker, namely BGE-M3 for our evaluation. We compute the reranker score across each cited passage  $p_i^j$  included in the long-form answer along with the user query  $q_i$ .
- **5. Answer Overlap:** Existing open-domain question answering datasets such as Natural Questions

- (Kwiatkowski et al., 2019), HotpotQA (Yang et al., 2018) or ELI5 (Fan et al., 2019) all include a human-labeled answer, assisting in evaluation using text overlap metrics such as Exact Match (EM) or F1. However, user queries in RAG systems potentially generate long-form answers, requiring metrics such as SacreBLEU or ROUGE-L to evaluate text-generation tasks. Automatic metrics are fairly quick and cheap to compute. For this reason, we include two metrics, SacreBLEU and ROUGE-L, for evaluating the RAG-generated answer. As we do not have human-labeled answers in MIRAGE-BENCH, we consider the GPT-4 generated answer as the gold truth for evaluation.
- **6. Answer overlap (LLM-measured):** To capture semantic overlap between answers, we use the Llama-3 (8B) model as the judge for evaluation in a pointwise setup, where the LLM as a judge outputs a score between 1 to 5.
- 7. Fluency (LLM-measured): Fluency measures for grammatical correctness and idiomatic word choices in long-form answer generation. Evaluating fluency in multilingual long-form generation answers is not straightforward. While existing techniques are available for English such as MAUVE (Pillutla et al., 2021), only a few models evaluate multilingual summarization (Clark et al., 2023). Inspired by recent works in G-EVAL (Liu et al., 2023), we evaluate fluency using open-source LLM such as Llama-3 (8B) as the judge. Our reason for choosing open-source models lies in reducing the expense, of running an expensive proprietary LLM such as GPT-4. Our LLM as a judge setup outputs a score between 1 to 5.

## **B** Baselines: Additional Details

In this section, we briefly describe each of the 19 multilingual-focused models utilized in our MI-RAGE-BENCH evaluation experiments:

- 1. **GPT-3.5-turbo:** (OpenAI, 2023) is evaluated using the Azure OpenAI service. <sup>10</sup> We set the temperature parameter to 0.1 for a deterministic output. It utilizes the cl100k\_base BPE-based tokenizer in the tiktoken<sup>11</sup> repository.
- 2. **GPT-4:** (OpenAI, 2023) is also evaluated using the Azure OpenAI service. We use a temperature setting of 0.1 for a deterministic output and the cl100k\_base BPE-based tokenizer.

<sup>&</sup>lt;sup>10</sup>https://learn.microsoft.com/en-us/azure/ai-services/openai/

<sup>11</sup>https://github.com/openai/tiktoken

- 3. **GPT-40:** (OpenAI, 2023) is also evaluated using the Azure OpenAI service. We use a temperature setting of 0.1 for a deterministic output and the o200k\_base BPE-based tokenizer.
- 4. **Mistral-7B-Instruct-v0.2** (Jiang et al., 2023) is the v0.2 of the instruct-version model containing 7B parameters. <sup>12</sup> It is an English-centric model, i.e., not instruction fine-tuned with any multilingual data. We used the multiple GPU inference using the v11m repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 5. **Mistral-7B-Instruct-v0.3** (Jiang et al., 2023) is an extension of the instruct-version 0.2 model containing 7B parameters.<sup>13</sup> We set inference parameters similar to the previous model.
- 6. **Mixtral-8x7B-Instruct-v0.1** (Jiang et al., 2024) is a pretrained generative sparse Mixture of Experts (MoE), containing 8x7B parameters. It has been pretrained in 5 languages including English, French, Italian, German, and Spanish. As the model is computationally not feasible to evaluate due to resource constraints, We use the model API endpoint available in the Anyscale platform (https://www.anyscale.com/), with a temperature setting of 0.1.
- 7. **Mixtral-8x22B-Instruct-v0.1** (Jiang et al., 2024) is a pretrained generative sparse Mixture of Experts (MoE), containing 8x22B parameters. Similar to before, it has pretrained on 5 languages including English, French, Italian, German, and Spanish. <sup>15</sup> We utilize the model API endpoint available in the Anyscale platform (https://www.anyscale.com/), with a temperature setting of 0.1.
- 8. **Command R** is developed keeping RAG in mind and officially supports 11 languages: Arabic, Brazilian, Portuguese, English, French, German, Italian, Japanese, Korean, Chinese, and Spanish. The model contains 35 billion parameters. We utilize the model API available in the Cohere platform (https://cohere.com/), with a temperature setting of 0.1, and using the chat template format.
- 9. **Command R+** is also developed keeping RAG in mind and officially supports 10 languages:

- English, French, Spanish, Italian, German, Brazilian Portuguese, Japanese, Korean, Arabic, and Chinese. The model contains 105 billion parameters. We utilize the model API available in the Cohere platform (https://cohere.com/), with a temperature setting of 0.1, and using the chat template format.
- 10. **Aya-23-35B** (Aryabumi et al., 2024) is an instruction fine-tuned model with highly advanced multilingual capabilities. The model officially supports 23 languages: Arabic, Chinese (simplified & traditional), Czech, Dutch, English, French, German, Greek, Hebrew, Hindi, Indonesian, Italian, Japanese, Korean, Persian, Polish, Portuguese, Romanian, Russian, Spanish, Turkish, Ukrainian, and Vietnamese. The model contains 35 billion parameters. We utilize the model API available in the Cohere platform (https://cohere.com/), with a temperature setting of 0.1, and using the chat template format.
- 11. **Gemma 1.1 (2B) it** (Mesnard et al., 2024) is an instruction fine-tuned model trained using the RLHF method containing 2 billion parameters. <sup>19</sup> We used the multiple GPU inference using the vllm repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 12. **Gemma 1.1 (7B) it** (Mesnard et al., 2024) is an instruction fine-tuned model trained using the RLHF method containing 7 billion parameters. <sup>20</sup> We used the multiple GPU inference using the vllm repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 13. **Meta-Llama-3-8B-Instruct** (Dubey et al., 2024) is an English-only instruction fine-tuned model containing 8 billion parameters.<sup>21</sup> We used the multiple GPU inference using the v11m repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 14. **Meta-Llama-3-70B-Instruct** (Dubey et al., 2024) is an instruction fine-tuned model containing 70B parameters.<sup>22</sup> As the model is computationally not feasible to evaluate due to resource constraints, We use the model API endpoint available in the Anyscale platform

<sup>&</sup>lt;sup>12</sup>mistralai/Mistral-7B-Instruct-v0.2

<sup>&</sup>lt;sup>13</sup>mistralai/Mistral-7B-Instruct-v0.3

<sup>&</sup>lt;sup>14</sup>mistralai/Mixtral-8x7B-Instruct-v0.1

<sup>&</sup>lt;sup>15</sup>mistralai/Mixtral-8x22B-Instruct-v0.1

<sup>&</sup>lt;sup>16</sup>CohereForAI/c4ai-command-r-v01

<sup>&</sup>lt;sup>17</sup>CohereForAI/c4ai-command-r-plus

<sup>&</sup>lt;sup>18</sup>CohereForAI/aya-23-35B

<sup>&</sup>lt;sup>19</sup>google/gemma-1.1-2b-it

<sup>&</sup>lt;sup>20</sup>google/gemma-1.1-7b-it

<sup>&</sup>lt;sup>21</sup>meta-llama/Meta-Llama-3-8B-Instruct

<sup>&</sup>lt;sup>22</sup>meta-llama/Meta-Llama-3-70B-Instruct

(https://www.anyscale.com/), with a temperature setting of 0.1.

- 15. **Phi-3** (**mini**) (Abdin et al., 2024) is an English-focused instruction fine-tuned model trained model containing 3.8 billion parameters.<sup>23</sup> We used the multiple GPU inference using the vllm repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 16. **Phi-3** (small) (Abdin et al., 2024) is a multilingual instruction fine-tuned model trained model containing 8 billion parameters.<sup>24</sup> There is no available information on the number of languages covered by the model. We used the multiple GPU inference using the vllm repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 17. **Phi-3** (**medium**) (Abdin et al., 2024) is a multilingual instruction fine-tuned model trained model containing 14 billion parameters. There is no available information on the number of languages covered by the model. We used the multiple GPU inference using the vllm repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 18. **Qwen2-1.5B-Instruct** (Yang et al., 2024a) is an English-focused instruction fine-tuned model trained model containing 1.5 billion parameters. <sup>26</sup> We used the multiple GPU inference using the vllm repository (Kwon et al., 2023). We set the temperature parameter to 0.1.
- 19. **Qwen2-7B-Instruct** (Yang et al., 2024a) is an English-focused instruction fine-tuned model trained model containing 7 billion parameters.<sup>27</sup> We used the multiple GPU inference using the vllm repository (Kwon et al., 2023). We set the temperature parameter to 0.1.

## C MIRAGE Fine-tuning Details

For multilingual RAG fine-tuning, we use teacher models to *distill* synthetic knowledge directly within smaller open-sourced models. We first generate RAG outputs on the MIRAGE-BENCH training dataset using three high-performing teacher models: (i) GPT-4o, (ii) Llama-3 (70B), and (iii) Mixtral (8x22B), and generate RAG output for

queries in MIRAGE-BENCH training dataset using only relevant passages, i.e., without distracting the model with information from non-relevant passages. We filter out the teacher model responses and curate them to create the training dataset.

Next, using supervised fine-tuning (SFT) with LoRA (Hu et al., 2022), we fine-tune two opensourced models: (i) Llama-3 (8B) and (ii) Mistralv0.2 (7B). Our hyperparameter choices are listed in Table 6. We use PEFT (Mangrulkar et al., 2022) and the alignment-handbook<sup>28</sup> (Tunstall et al., 2023) for supervised LoRA fine-tuning. We finetune four variants of models: (i) Mistral-v0.2 (7B) distilled using GPT-40 as a teacher, (ii) Mistralv0.2 (7B) distilled using Mixtral (8x22B) and itself as a teacher, (iii) Llama-3 (8B) distilled using GPT-4o as a teacher, and (iv) Llama-3 (8B) distilled using Llama-3 (70B) and itself as a teacher. After fine-tuning, first all heuristic features are computed, using the already trained regression model (using the baselines) is used to compute inference for the fine-tuned models and compared against upper-bound baselines, GPT-40, Llama-3 (70B), and Mixtral (8x22B) and lower-bound baselines, Mistral-v0.2 (7B) and Llama-3 (8B).

## **D** Extending MIRAGE Evaluation

As a holdout experiment, we evaluate newer versions of models, (i) Llama-3.1 series (Dubey et al., 2024): Llama-3.1 (8B)<sup>29</sup> and Llama-3.1 (70B)<sup>30</sup> instruct versions, and (ii) Gemma-2 series (Team et al., 2024): Gemma-2 (9B)<sup>31</sup> and Gemma-2 (27B)<sup>32</sup> instruct versions. For both models, we used the API versions of the model provided by NVIDIA (https://build.nvidia.com/) by setting the temperature parameter to 0.1. The maximum sequence length of Gemma-2 models is 4096 tokens. **Experimental results.** From Figure 8, we observe

that the Gemma-2 (27B) and Llama-3.1 (70B) are strong baselines, by achieving an overall rank of 4 and 5 in the MIRAGE-BENCH dataset. Gemma-2 (27B) improves the previously best Gemma-1.1 (7B) by 13 ranks, whereas Llama-3.1 (70B) continues to underperform the best Llama-3 (70B) by 2 ranks. These results indicate newer models are improving, as reported using the surrogate judge on the synthetic MIRAGE-BENCH leaderboard.

<sup>&</sup>lt;sup>23</sup>microsoft/Phi-3-mini-128k-instruct

<sup>&</sup>lt;sup>24</sup>microsoft/Phi-3-small-8k-instruct

<sup>&</sup>lt;sup>25</sup>microsoft/Phi-3-medium-128k-instruct

<sup>&</sup>lt;sup>26</sup>Qwen/Qwen2-1.5B-Instruct

<sup>&</sup>lt;sup>27</sup>Qwen/Qwen2-7B-Instruct

<sup>&</sup>lt;sup>28</sup>https://github.com/huggingface/alignment-handbook

<sup>&</sup>lt;sup>29</sup>meta-llama/Meta-Llama-3.1-8B-Instruct

<sup>&</sup>lt;sup>30</sup>meta-llama/Meta-Llama-3.1-70B-Instruct

<sup>&</sup>lt;sup>31</sup>google/gemma-2-9b-it

<sup>&</sup>lt;sup>32</sup>google/gemma-2-27b-it

Lang.	Mean	95% CI	Lang.	Mean	95% CI	Lang.	Mean	95% CI
ar	0.916	-0.15 / +0.07	bn	0.937	-0.17 / +0.06	de	0.939	-0.14 / +0.05
en	0.971	-0.07 / +0.03	es	0.844	-0.12 / +0.09	fa	0.944	-0.22 / +0.05
fi	0.957	-0.07 / +0.04	fr	0.861	-0.15 / +0.09	hi	0.858	-0.26 / +0.13
id	0.793	-0.17 / +0.12	ja	0.892	-0.13 / +0.08	ko	0.941	-0.13 / +0.06
ru	0.968	-0.11 / +0.03	sw	0.973	-0.06 / +0.03	te	0.929	-0.16 / +0.07
th	0.902	-0.12 / +0.09	yo	0.709	-0.22 / +0.16	zh	0.954	-0.09 / +0.05

Table 5:  $\bar{R}^2$  mean scores with 95% confidence interval with bootstrapping across all languages in MIRAGE-BENCH.

Hyper-parameter	Choice
Attention	FlashAttention-2 (Dao, 2024)
Batch Size	32
Epochs	1
Learning Rate	2e-4
Max Sequence Length	6144
Lora rank $(r)$	16
Lora alpha $(\alpha)$	16
Lora dropout	0.05
Lora Modules	[q_proj, k_proj, v_proj,
	<pre>o_proj, gate_proj, up_proj, down_proj]</pre>

Table 6: Hyperparameter settings set during supervised fine-tuning of Mistral-v0.2 (7B) and Llama-3 (8B) on the MIRAGE-BENCH training dataset.

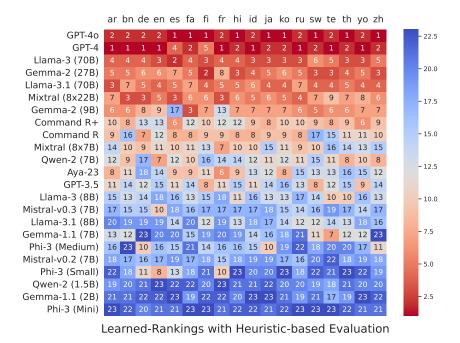


Figure 8: Approximate rankings using heuristic features including the newer models, Llama-3.1 (Dubey et al., 2024) and Gemma-2 (Team et al., 2024) on MIRAGE-BENCH dataset across all 18 languages. Gemma-2 (27B) and Llama-3.1 (70B) achieve a strong rank of 4 and 5 respectively in the MIRAGE-BENCH evaluation dataset.

	,		Deligali (DII)	(110)		Commun (ac)	(1)	•	(III) IIISIII (CII)	(en)		(ca) memodic	(i)	
Model Name Mc	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI
1. GPT-4 2.8	2.838 -0.2	-0.22/+0.28   1. GPT-4o	1. GPT-40	3.104	-0.31/+0.33   1. GPT-4o	1. GPT-40	2.644	-0.21/+0.26   1. GPT-40	1. GPT-40	2.311	-0.22/+0.21   1. GPT-40	1. GPT-40	2.280	-0.30/+0.32
2. GPT-40 2.5	2.549 -0.2	-0.27/+0.24   2. GPT-4	2. GPT-4	2.881	-0.32/+0.34   2. GPT-4	2. GPT-4	2.498	-0.27/+0.26 2. GPT-4	2. GPT-4	2.235	-0.27/+0.30 2. GPT-4	2. GPT-4	1.230	-0.28/+0.27
3. Llama-3 (70B) 1.2	271 -0.1	.   05.0+/61	-0.19/+0.20   3. Llama-3 (70B)	1.290	-0.28/+0.26	3. Mixtral (8x22B)	1.002	-0.21/+0.23	3. Llama-3 (70B)	1.082	-0.22/+0.19	-0.22/+0.19 3. Llama-3 (70B)	1.225	-0.17/+0.20
4. Mixtral (8x22B) 1.0	1.074 -0.2	23/+0.27	0.23/+0.27   4. Mixtral (8x22B)	1.266	-0.22/+0.24	4. Llama-3 (70B)	0.921	-0.23/+0.22	4. Mixtral (8x22B)	0.689	-0.18/+0.18	-0.18/+0.18   4. Mixtral (8x22B)	0.832	-0.21/+0.21
5. Command R 0.6	0.696 -0.1	19/+0.23	-0.19/+0.23   5. Qwen2 (7B)	0.486	-0.20/+0.22	5. Mixtral (8x7B)	0.481	-0.25/+0.23	5. Qwen2 (7B)	0.256	-0.24/+0.24 5. Aya-23	5. Aya-23	0.485	-0.24/+0.22
6. Command R+ 0.4	0.476 -0.1	19/+0.24	-0.19/+0.24   6. Command R+	0.373	-0.18/+0.23	<ol><li>Command R</li></ol>	0.213	-0.21/+0.20	6. Phi-3 (small)	0.198	-0.24/+0.26	-0.24/+0.26   6. Mixtral (8x7B)	0.476	-0.21/+0.25
7. Aya-23 0.4	.450 -0.1	. 02.0+/61	0.19/+0.20   7. Mixtral (8x7B)	0.112	-0.24/+0.22	7. Phi-3 (small)	0.148	-0.21/+0.20	7. Mixtral (8x7B)	0.119	-0.23/+0.26	0.23/+0.26   7. Command R	0.402	-0.18/+0.21
8. GPT-3.5 0.3	0.344 -0.2	-0.28/+0.28   8. Aya-23	8. Aya-23	0.039	-0.21/+0.19	8. Aya-23	0.057	-0.21/+0.18	8. Command R	0.011	-0.19/+0.19	-0.19/+0.19 8. Command R+	0.284	-0.20/+0.18
9. Qwen2 (7B) 0.2	0.293 -0.2	23/+0.21	-0.23/+0.21   9. Gemma-1.1 (7B)	-0.141	-0.20/+0.21	9. Phi-3 (medium)	-0.027	-0.20/+0.18	9. Aya-23	-0.059	-0.23/+0.22 9. GPT-3.5	9. GPT-3.5	-0.029	-0.18/+0.22
<ol> <li>Mixtral (8x7B) 0.1</li> </ol>	0.139 -0.2	24/+0.28	-0.24/+0.28   10. Llama-3 (8B)	-0.143	-0.20/+0.19	10. GPT-3.5	-0.077	-0.18/+0.17	10. Command R+	-0.193	-0.24/+0.25	-0.24/+0.25   10. Phi-3 (small)	-0.044	-0.21/+0.22
11. Llama-3 (8B) -0.4	-0.407 -0.2	20/+0.24	-0.20/+0.24   11. GPT-3.5	-0.388	-0.27/+0.27	3.27/+0.27   11. Command R+	-0.216	-0.20/+0.19	0.20/+0.19   11. Mistral-v0.3 (7B)	-0.259	-0.20/+0.23	-0.20/+0.23   11. Phi-3 (medium)	-0.056	-0.20/+0.20
12. Gemma-1.1 (7B) -0.4	-0.444 -0.1	19/+0.18	-0.19/+0.18   12. Mistral-v0.3 (7B)	-0.891	-0.24/+0.25	3.24/+0.25   12. Llama-3 (8B)	-0.224	-0.16/+0.17	0.16/+0.17   12. Mistral-v0.2 (7B)	-0.274	-0.23/+0.23	-0.23/+0.23   12. Qwen2 (7B)	-0.095	-0.20/+0.19
13. Phi-3 (medium) -0.6	-0.635 -0.2	26/+0.31	0.26/+0.31   13. Command R	-0.908	-0.16/+0.15	0.16/+0.15   13. Mistral-v0.2 (7B)	-0.331	-0.21/+0.20	13. GPT-3.5	-0.314	-0.25/+0.25	0.25/+0.25   13. Llama-3 (8B)	-0.247	-0.23/+0.23
14. Mistral-v0.3 (7B) -0.9	-0.972 -0.2	25/+0.23	-0.25/+0.23   14. Qwen2 (1.5B)	-1.038	-0.28/+0.30	3.28/+0.30   14. Qwen2 (7B)	-0.332	-0.20/+0.23	0.20/+0.23   14. Phi-3 (medium)	-0.330	-0.21/+0.24	-0.21/+0.24   14. Mistral-v0.3 (7B)	-0.338	-0.22/+0.20
15. Mistral-v0.2 (7B) -1.0	-1.076 -0.2	23/+0.21	-0.23/+0.21   15. Gemma-1.1 (2B)	-1.111	-0.27/+0.23	0.27/+0.23   15. Mistral-v0.3 (7B)	-0.363	-0.20/+0.20	0.20/+0.20   15. Llama-3 (8B)	-0.444	-0.19/+0.15	-0.19/+0.15   15. Mistral-v0.2 (7B)	-0.576	-0.22/+0.25
16. Qwen2 (1.5B) -1.1	1.180 -0.2	25/+0.23	0.25/+0.23   16. Mistral-v0.2 (7B)	-1.151	-0.25/+0.22	3.25/+0.22   16. Phi-3 (mini)	-1.262	-0.24/+0.22	0.24/+0.22   16. Gemma-1.1 (7B)	-0.764	-0.19/+0.20	0.19/+0.20   16. Gemma-1.1 (7B)	-1.023	-0.30/+0.26
17. Gemma-1.1 (2B) -1.6	-1.611 -0.1	17/+0.18	-0.17/+0.18   17. Phi-3 (small)	-1.201	-0.19/+0.16	0.19/+0.16   17. Qwen2 (1.5B)	-1.623	-0.31/+0.28	0.31/+0.28   17. Phi-3 (mini)	-1.123	-0.23/+0.20	-0.23/+0.20   17. Phi-3 (mini)	-1.412	-0.31/+0.29
18. Phi-3 (small) -1.6	1.612 -0.2	22/+0.24	-0.22/+0.24   18. Phi-3 (mini)	-1.269	-0.18/+0.15	0.18/+0.15   18. Gemma-1.1 (2B)	-1.716	-0.23/+0.21	0.23/+0.21   18. Gemma-1.1 (2B)	-1.308	-0.29/+0.27	-0.29/+0.27   18. Qwen2 (1.5B)	-1.476	-0.29/+0.28
19. Phi-3 (mini) -2.1	2.194 -0.2	23/+0.26	-0.23/+0.26   19. Phi-3 (medium)	-1.312	-0.23/+0.20	0.23/+0.20   19. Gemma-1.1 (7B)	-1.791	-0.26/+0.22	-0.26/+0.22   19. Qwen2 (1.5B)	-1.832	-0.26/+0.24	-0.26/+0.24   19. Gemma-1.1 (2B)	-1.919	-0.32/+0.24

Fars	Farsi (fa)		Finnish (fi)	sh (fi)		French (fr)	h (fr)		Hind	Hindi (hi)		Indonesian (id)	ian (id)	
Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI
1. GPT-40	2.951	-0.24/+0.27   1. GPT-40	1. GPT-40	2.623	-0.22/+0.24   1. GPT-4	1. GPT-4	2.795	-0.35/+0.40   1. GPT-4	1. GPT-4	3.065	-0.36/+0.31   1. GPT-4	1. GPT-4	2.207	-0.30/+0.28
2. GPT-4	2.639	-0.27/+0.33	0.27/+0.33   2. Llama-3 (70B)	1.616	-0.20/+0.19	2. GPT-40	2.476	-0.35/+0.32	-0.35/+0.32 2. GPT-40	2.958	-0.25/+0.26 2. GPT-4o	2. GPT-40	2.074	-0.30/+0.34
3. Llama-3 (70B)	1.533	-0.17/+0.16 3. GPT-4	3. GPT-4	1.552	-0.30/+0.38	3. Llama-3 (70B)	1.070	-0.22/+0.22	0.22/+0.22   3. Llama-3 (70B)	1.198	-0.21/+0.20	-0.21/+0.20 3. Llama-3 (70B)	1.296	-0.21/+0.21
<ol> <li>Mixtral (8x22B)</li> </ol>	1.257	-0.21/+0.24	0.21/+0.24   4. Mixtral (8x22B)	1.047	-0.25/+0.24	<ol> <li>Mixtral (8x22B)</li> </ol>	1.049	-0.24/+0.18	4. Mixtral (8x22B)	1.066	-0.23/+0.23	0.23/+0.23   4. Mixtral (8x22B)	0.848	-0.25/+0.26
5. Command R+	0.748	-0.21/+0.22	0.21/+0.22   5. GPT-3.5	0.513	-0.20/+0.23	5. Aya-23	0.507	-0.22/+0.23	5. Command R	0.812	-0.21/+0.22	0.21/+0.22   5. Command R	0.337	-0.18/+0.21
6. Command R	0.699	-0.19/+0.20	0.19/+0.20   6. Command R	0.452	-0.25/+0.19	6. Mixtral (8x7B)	0.504	-0.25/+0.26 6. Aya-23	6. Aya-23	0.646	-0.22/+0.27	-0.22/+0.27   6. Command R+	0.187	-0.16/+0.19
7. Qwen2 (7B)	0.423	-0.26/+0.26	-0.26/+0.26   7. Command R+	0.302	-0.21/+0.19	7. Command R	0.332	-0.18/+0.21	-0.18/+0.21 7. Command R+	0.586	-0.19/+0.21 7. GPT-3.5	7. GPT-3.5	0.083	-0.22/+0.21
8. Aya-23	0.411	-0.20/+0.23	0.20/+0.23   8. Mixtral (8x7B)	0.242	-0.27/+0.27	8. GPT-3.5	0.091	-0.20/+0.20	8. Mixtral (8x7B)	0.417	-0.25/+0.26	0.25/+0.26 8. Qwen2 (7B)	-0.007	-0.24/+0.25
<ol> <li>Mixtral (8x7B)</li> </ol>	0.395	-0.26/+0.31   9. Aya-23	9. Aya-23	0.187	-0.21/+0.23	<ol><li>Command R+</li></ol>	0.026	-0.26/+0.24	9. Qwen2 (7B)	0.041	-0.22/+0.21 9. Aya-23	9. Aya-23	-0.012	-0.21/+0.20
10. Llama-3 (8B)	-0.024	-0.26/+0.25	-0.26/+0.25   10. Phi-3 (medium)	-0.068	-0.26/+0.23	10. Phi-3 (small)	-0.058	-0.24/+0.26	-0.24/+0.26 10. Llama-3 (8B)	-0.032	-0.20/+0.25	-0.20/+0.25   10. Mixtral (8x7B)	-0.046	-0.27/+0.26
11. GPT-3.5	-0.091	-0.23/+0.27	0.23/+0.27   11. Llama-3 (8B)	-0.088	-0.19/+0.17	11. Qwen2 (7B)	-0.175	-0.27/+0.27	11. GPT-3.5	-0.170	-0.22/+0.24	-0.22/+0.24   11. Llama-3 (8B)	-0.274	-0.17/+0.19
12. Gemma-1.1 (7B)	-0.671	-0.22/+0.20	0.22/+0.20   12. Qwen2 (7B)	-0.246	-0.24/+0.22	12. Phi-3 (medium)	-0.385	-0.23/+0.19	12. Mistral-v0.3 (7B)	-0.519	-0.22/+0.28	0.22/+0.28   12. Mistral-v0.3 (7B)	-0.321	-0.22/+0.27
13. Mistral-v0.3 (7B)	-0.763	-0.27/+0.23	0.27/+0.23   13. Mistral-v0.3 (7B)	-0.543	-0.17/+0.19	.17/+0.19   13. Mistral-v0.2 (7B)	-0.442	-0.21/+0.22	-0.21/+0.22   13. Phi-3 (medium)	-0.620	-0.31/+0.28	-0.31/+0.28   13. Gemma-1.1 (7B)	-0.466	-0.23/+0.21
14. Mistral-v0.2 (7B)	-1.127	-0.22/+0.23	0.22/+0.23   14. Mistral-v0.2 (7B)	-0.756	-0.17/+0.20	14. Mistral-v0.3 (7B)	-0.457	-0.22/+0.21	14. Mistral-v0.2 (7B)	-0.873	-0.18/+0.19	0.18/+0.19   14. Phi-3 (medium)	-0.529	-0.21/+0.23
15. Gemma-1.1 (2B)	-1.258	-0.23/+0.18	0.23/+0.18   15. Qwen2 (1.5B)	-0.975	-0.22/+0.24	15. Llama-3 (8B)	-0.653	-0.17/+0.18	15. Gemma-1.1 (2B)	-1.205	-0.27/+0.23	0.27/+0.23   15. Mistral-v0.2 (7B)	-0.591	-0.25/+0.19
16. Phi-3 (medium)	-1.368	-0.21/+0.24	-0.21/+0.24   16. Gemma-1.1 (7B)	-1.029	-0.19/+0.20	-0.19/+0.20   16. Phi-3 (mini)	-1.353	-0.27/+0.27	-0.27/+0.27   16. Gemma-1.1 (7B)	-1.257	-0.25/+0.27	-0.25/+0.27   16. Phi-3 (small)	-0.627	-0.26/+0.25
17. Qwen2 (1.5B)	-1.501	-0.21/+0.26	-0.21/+0.26   17. Phi-3 (small)	-1.316	-0.23/+0.23	17. Gemma-1.1 (7B)	-1.491	-0.24/+0.25	17. Qwen2 (1.5B)	-1.529	-0.25/+0.25	-0.25/+0.25   17. Qwen2 (1.5B)	-0.967	-0.27/+0.26
18. Phi-3 (small)	-1.623	-0.21/+0.18	0.21/+0.18   18. Gemma-1.1 (2B)	-1.420	-0.24/+0.23	18. Qwen2 (1.5B)	-1.861	-0.32/+0.28	18. Phi-3 (mini)	-2.192	-0.20/+0.22	.0.20/+0.22   18. Phi-3 (mini)	-1.514	-0.29/+0.26
19. Phi-3 (mini)	-2.629		-0.34/+0.28   19. Phi-3 (mini)	-2.092	-0.26/+0.26	-0.26/+0.26   19. Gemma-1.1 (2B)	-1.977	-0.24/+0.25	-0.24/+0.25   19. Phi-3 (small)	-2.394	-0.31/+0.27	-0.31/+0.27   19. Gemma-1.1 (2B)	-1.678	-0.25/+0.26

Table 7: Bradley-Terry logits using GPT-40 as a judge for all languages in MIRAGE-BENCH. Scores are computed using the Bradley-Terry model with 200 tournaments using a maximum of 100 randomly sampled queries. Mean scores and 95% confidence intervals are reported (repeated 200 times). A higher logit score indicates a better performance, therefore achieving a higher rank on MIRAGE-BENCH. Models are sorted in descending order of mean score for each language in MIRAGE-BENCH.

Japanesee (ja)	see (ja)		Korean (ko)	n (ko)		Russian (ru)	m (ru)		Swahili (sw)	i (sw)	
Model Name	Mean	95% CI	Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI
1. GPT-40	2.294	-0.24/+0.32	1. GPT-40	2.402	-0.34/+0.35   1. GPT-4	1. GPT-4	2.372	-0.26/+0.25   1. GPT-4o	1. GPT-40	2.966	-0.25/+0.28
2. GPT-4	2.173	-0.24/+0.24	2. GPT-4	2.158	-0.26/+0.31 2. GPT-40	2. GPT-40	2.208	-0.27/+0.25   2. GPT-4	2. GPT-4	2.038	-0.36/+0.40
3. Llama-3 (70B)	1.088	-0.22/+0.23	3. Llama-3 (70B)	1.391	-0.23/+0.21	3. Llama-3 (70B)	1.323	-0.20/+0.21	0.20/+0.21   3. Llama-3 (70B)	1.750	-0.22/+0.20
<ol> <li>Mixtral (8x22B)</li> </ol>	0.834	-0.18/+0.19	4. Mixtral (8x22B)	1.028	-0.24/+0.23	4. Mixtral (8x22B)	1.196	-0.24/+0.25	4. Mixtral (8x22B)	1.355	-0.19/+0.22
5. Command R	0.377	-0.20/+0.20	5. Command R	0.647	-0.20/+0.26	5. Mixtral (8x7B)	0.821	-0.26/+0.27	0.26/+0.27   5. GPT-3.5	1.006	-0.24/+0.24
6. Aya-23	0.290	-0.21/+0.19	6. Aya-23	0.645	-0.20/+0.20	.0.20/+0.20 6. Command R	0.687	-0.16/+0.17	6. Command R+	0.735	-0.19/+0.17
7. Qwen2 (7B)	0.269	-0.27/+0.27	7. Mixtral (8x7B)	0.617	-0.25/+0.20 7. Aya-23	7. Aya-23	0.276	-0.17/+0.20	0.17/+0.20   7. Mixtral (8x7B)	0.473	-0.23/+0.24
8. Phi-3 (medium)	0.225	-0.21/+0.25	8. Qwen2 (7B)	0.218	-0.25/+0.23 8. GPT-3.5	8. GPT-3.5	0.261	-0.25/+0.21	0.25/+0.21   8. Gemma-1.1 (7B)	0.385	-0.23/+0.23
<ol> <li>Mixtral (8x7B)</li> </ol>	-0.020	-0.27/+0.27	9. Command R+	0.026	-0.18/+0.17	9. Command R+	0.121	-0.19/+0.20   9. Aya-23	9. Aya-23	-0.042	-0.24/+0.24
10. Command R+	-0.044	-0.17/+0.21	10. Llama-3 (8B)	-0.041	-0.20/+0.20	.0.20/+0.20   10. Qwen2 (7B)	0.116	-0.28/+0.26	0.28/+0.26   10. Llama-3 (8B)	-0.057	-0.23/+0.20
11. Llama-3 (8B)	-0.099	-0.18/+0.18	11. Mistral-v0.3 (7B)	-0.321	-0.24/+0.24	.0.24/+0.24   11. Mistral-v0.3 (7B)	-0.074	-0.24/+0.24	0.24/+0.24   11. Qwen2 (7B)	-0.175	-0.30/+0.25
12. GPT-3.5	-0.292	-0.21/+0.20	12. GPT-3.5	-0.352	-0.21/+0.23	0.21/+0.23   12. Mistral-v0.2 (7B)	-0.187	-0.28/+0.23	0.28/+0.23   12. Phi-3 (medium)	-0.783	-0.25/+0.27
13. Gemma-1.1 (7B)	-0.418	-0.22/+0.22	13. Mistral-v0.2 (7B)	-0.467	-0.27/+0.27	13. Llama-3 (8B)	-0.401	-0.19/+0.21	0.19/+0.21   13. Mistral-v0.3 (7B)	-0.804	-0.27/+0.25
14. Mistral-v0.3 (7B)	-0.498	-0.24/+0.25	14. Gemma-1.1 (7B)	-0.513	-0.23/+0.26	14. Phi-3 (small)	-1.026	-0.24/+0.28	14. Command R	-1.008	-0.23/+0.23
15. Mistral-v0.2 (7B)	-0.599	-0.22/+0.25	15. Phi-3 (medium)	-1.046	-0.24/+0.27	.0.24/+0.27   15. Qwen2 (1.5B)	-1.135	-0.26/+0.22	0.26/+0.22   15. Qwen2 (1.5B)	-1.077	-0.26/+0.27
16. Phi-3 (small)	-0.947	-0.30/+0.27	16. Qwen2 (1.5B)	-1.162	-0.27/+0.22	16. Gemma-1.1 (2B)	-1.216	-0.23/+0.26	0.23/+0.26   16. Mistral-v0.2 (7B)	-1.356	-0.29/+0.24
17. Qwen2 (1.5B)	-1.471	-0.27/+0.26	17. Gemma-1.1 (2B)	-1.263	-0.25/+0.27	17. Gemma-1.1 (7B)	-1.469	-0.21/+0.23	0.21/+0.23   17. Gemma-1.1 (2B)	-1.448	-0.27/+0.23
18. Phi-3 (mini)	-1.548	-0.26/+0.24	18. Phi-3 (small)	-1.864	-0.24/+0.23	-0.24/+0.23   18. Phi-3 (medium)	-1.758	-0.17/+0.18	0.17/+0.18   18. Phi-3 (small)	-1.937	-0.30/+0.35
19. Gemma-1.1 (2B)	-1.615	-0.32/+0.23	19. Phi-3 (mini)	-2.104	-0.22/+0.24	-0.22/+0.24   19. Phi-3 (mini)	-2.115	-0.25/+0.25	0.25/+0.25   19. Phi-3 (mini)	-2.019	-0.30/+0.29

Telugu (te)	u (te)		Thai (th)	(th)		Yoruba (yo)	a (yo)		Chine	Chinese (zh)	
Model Name	Mean	95% CI	Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI	95% CI   Model Name	Mean	95% CI
1. GPT-4	3.112	-0.34/+0.33	1. GPT-40	2.606	-0.27/+0.27   1. GPT-40	1. GPT-40	2.606	-0.27/+0.27   1. GPT-4o	1. GPT-40	2.352	-0.25/+0.27
2. GPT-40	3.083	-0.26/+0.32	2. GPT-4	2.348	-0.27/+0.30 2. GPT-4	2. GPT-4	2.348	-0.27/+0.30 2. GPT-4	2. GPT-4	2.173	-0.31/+0.31
3. Llama-3 (70B)	1.229	-0.23/+0.22	3. Llama-3 (70B)	1.405	-0.20/+0.21	-0.20/+0.21 3. Llama-3 (70B)	1.405	-0.20/+0.21	0.20/+0.21   3. Mixtral (8x22B)	0.987	-0.23/+0.24
4. Gemma-1.1 (7B)	1.046	-0.26/+0.21	4. Mixtral (8x22B)	1.083	-0.28/+0.25	-0.28/+0.25 4. Mixtral (8x22B)	1.083	-0.28/+0.25	-0.28/+0.25 4. Llama-3 (70B)	0.950	-0.20/+0.20
5. Command R+	0.566	-0.21/+0.23	5. Qwen2 (7B)	0.855	-0.26/+0.27	5. Qwen2 (7B)	0.855	-0.26/+0.27	-0.26/+0.27   5. Command R	0.502	-0.20/+0.20
<ol><li>Mixtral (8x22B)</li></ol>	0.466	-0.22/+0.25	6. Command R+	0.513	-0.22/+0.19	6. Command R+	0.513	-0.22/+0.19	6. Qwen2 (7B)	0.455	-0.26/+0.26
7. Llama-3 (8B)	0.255	-0.19/+0.21	7. Llama-3 (8B)	0.421	-0.28/+0.24	-0.28/+0.24 7. Llama-3 (8B)	0.421	-0.28/+0.24	-0.28/+0.24 7. Command R+	0.344	-0.21/+0.19
8. Qwen2 (7B)	0.1111	-0.26/+0.27	8. Gemma-1.1 (7B)	-0.045	-0.19/+0.23	8. Gemma-1.1 (7B)	-0.045	-0.19/+0.23	8. Phi-3 (medium)	0.125	-0.20/+0.19
9. GPT-3.5	0.074	-0.23/+0.25	9. Command R	-0.135	-0.19/+0.20	-0.19/+0.20 9. Command R	-0.135	-0.19/+0.20	9. Aya-23	0.106	-0.22/+0.22
10. Aya-23	-0.203	-0.24/+0.22	10. Mixtral (8x7B)	-0.201	-0.28/+0.28	-0.28/+0.28   10. Mixtral (8x7B)	-0.201	-0.28/+0.28	-0.28/+0.28   10. GPT-3.5	-0.163	-0.23/+0.21
11. Command R	-0.404	-0.26/+0.23	11. GPT-3.5	-0.233	-0.30/+0.28	11. GPT-3.5	-0.233	-0.30/+0.28	11. Mixtral (8x7B)	-0.177	-0.23/+0.21
<ol> <li>Mixtral (8x7B)</li> </ol>	-0.739	-0.25/+0.28	12. Mistral-v0.3 (7B)	-0.332	-0.23/+0.31	12. Mistral-v0.3 (7B)	-0.332	-0.23/+0.31	-0.23/+0.31   12. Llama-3 (8B)	-0.200	-0.20/+0.22
13. Gemma-1.1 (2B)	-1.045	-0.20/+0.24	13. Aya-23	-0.365	-0.22/+0.19   13. Aya-23	13. Aya-23	-0.365	-0.22/+0.19	-0.22/+0.19   13. Mistral-v0.3 (7B)	-0.261	-0.25/+0.25
14. Phi-3 (medium)	-1.110	-0.20/+0.20	14. Mistral-v0.2 (7B)	-1.008	-0.25/+0.25	14. Mistral-v0.2 (7B)	-1.008	-0.25/+0.25	0.25/+0.25   14. Mistral-v0.2 (7B)	-0.584	-0.21/+0.20
15. Qwen2 (1.5B)	-1.119	-0.20/+0.16	15. Phi-3 (medium)	-1.040	-0.22/+0.19	15. Phi-3 (medium)	-1.040	-0.22/+0.19	0.22/+0.19   15. Phi-3 (small)	-0.883	-0.24/+0.26
16. Mistral-v0.3 (7B)	-1.194	-0.19/+0.18	16. Gemma-1.1 (2B)	-1.103	-0.24/+0.22	.0.24/+0.22   16. Gemma-1.1 (2B)	-1.103	-0.24/+0.22	-0.24/+0.22   16. Qwen2 (1.5B)	-0.919	-0.22/+0.24
17. Phi-3 (small)	-1.275	-0.18/+0.18	17. Qwen2 (1.5B)	-1.255	-0.38/+0.32	17. Qwen2 (1.5B)	-1.255	-0.38/+0.32	0.38/+0.32   17. Gemma-1.1 (2B)	-1.265	-0.26/+0.31
18. Mistral-v0.2 (7B)	-1.348	-0.20/+0.18	18. Phi-3 (small)	-1.740	-0.22/+0.22	18. Phi-3 (small)	-1.740	-0.22/+0.22	-0.22/+0.22   18. Phi-3 (mini)	-1.503	-0.26/+0.28
19. Phi-3 (mini)	-1.504	-0.26/+0.19	19. Phi-3 (mini)	-1.774	-0.24/+0.22	-0.24/+0.22   19. Phi-3 (mini)	-1.774	-0.24/+0.22	-0.24/+0.22   19. Gemma-1.1 (7B)	-2.037	-0.38/+0.32

Table 8: Bradley-Terry logits using GPT-40 as a judge for all languages in MIRAGE-BENCH. Scores are computed using the Bradley-Terry model with 200 tournaments using a maximum of 100 randomly sampled queries. Mean scores and 95% confidence intervals are reported (repeated 200 times). A higher logit score indicates a better performance, therefore achieving a higher rank on MIRAGE-BENCH. Models are sorted in descending order of mean score for every language in MIRAGE-BENCH.



Figure 9: Lollipop plots denoting the average heuristic-based feature scores achieved by baselines in MIRAGE-BENCH for all eleven heurisitc-based features. x-axis denotes the languages in MIRAGE-BENCH; whereas y-axis plots every heuristic feature value. Multiple LLMs in the same family are represented as a single color lollipop (multiple circles).

```
Ouestion:
What was the first newspaper ever printed in the U.K.?
[36897421#2]" Lögberg-Heimskringla - The very first newspaper to be published in North America
by the Icelandic immigrant population was handwritten by Jon Gudmundsson in 1876 ...
"[1965416#2]" The New York Times Magazine - Its first issue was published on September 6, 1896,
and contained the first photographs ever printed in the newspaper...
"[662134#6]" Letterpress printing - Letterpress printing was introduced in Canada in 1752 in
Halifax, Nova Scotia by John Bushell in the newspaper format. This paper was named the Halifax
Gazette and became Canada's first newspaper ...
"[22112840#15]" Newspaper - The emergence of the new media in the 17th century has to be
seen in close connection with the spread of the printing press from which the publishing press
derives its name....
Instruction:
Provide an answer to the question using the information provided in contexts written in
{{language}}. Additionally, provide a step-by-step explanation of your reasoning, demonstrating
how you arrived at your answer in {{language}}. Cite parts of your reasoning within brackets []
using the IEEE format based on the provided contexts.
Please respond in {{language}} using the format: ##Reason: {reason} ##Answer: {answer}.
```

Figure 10: Prompt template for all baseline models for multilingual RAG generation for queries across all languages in MIRAGE-BENCH. We include the language-specific query in MIRAGE-BENCH under "Question:". Next, we concatenate both relevant and non-relevant passages (randomly shuffled and truncated at maximum length) and place them under "Contexts:". Lastly, we provide our instruction in English asking the model to generate a response in the required language under the placeholder "{{language}}". The example above is shown for a query in English (en) from MIRAGE-BENCH, where contexts are truncated ( ... ) for demonstration purposes.

```
You are an AI assistant. In the following task, you are given a Question, a RAG application's
response, and a Ground-truth Answer referred to as 'Label' in {{language}}. Assess how well the
RAG application's response aligns with the Label, using the grading rubric below:
1: The response is not aligned with the Label or is off-topic; it includes hallucination.
2: The response admits it cannot provide an answer or lacks context; it is honest.
3: The response is relevant but contains notable discrepancies or inaccuracies.
4: The response is acceptable and sufficient but not exhaustive.
5: The response is fully accurate and comprehensive, based on the Label.
Treat the Label as the definitive answer. Present your final score in the format: "[[score]]",
followed by your justification in English. Example:
Score: [[3]] Justification: The response partially aligns with the Label but with some
discrepancies.
Question in {{language}}:
{{Question}}
Label in {{language}}:
{{Label}}
RAG Application Response in {{language}}:
{{Response}}
Treat the Label as the definitive answer. Present your final score in the format: "[[score]]",
followed by your justification in English.
```

Figure 11: Prompt template used by Llama-3 (8B) model as a judge to evaluate the answer overlap heuristic feature. We include a grading rubric within the prompt template. {{Label}} is a placeholder for the gold truth answer provided using the GPT-4; {{language}} is a placeholder for the target language; {{Question}} is a placeholder for the MIRAGE-BENCH query; {{Documents}} is a placeholder for both MIRAGE-BENCH relevant and non-relevant passages concatenated together; {{Response}} is a placeholder for RAG model output.

You will be given one summary written for a question and documents from Wikipedia in {{language}}. Your task is to rate the summary on one metric. Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.

#### Evaluation Criteria:

Fluency (1-5) - the collective quality of all sentences. We align this dimension with the DUC quality question of structure and fluency whereby "the summary should be well-structured and well-organized. The summary should not just be a heap of related information, but should build from sentence to sentence to a coherent body of information about a topic."

#### Evaluation Steps:

- 1. Read the question and Wikipedia documents in {{language}} carefully and identify the main topic and key points.
- 2. Read the summary and check whether it answers the question. Check if the summary covers the main topic and key points required to answer the question, and if it presents them in a clear and logical order.
- 3. Assign a rating for fluency on a scale of 1 to 5 and provide an explanation, where 1 is the lowest and 5 is the highest based on the Evaluation Criteria.

```
Example:
Question in {{language}}:
{{Question}}

Documents in {{language}}:
{{Documents}}

Summary:
{{Summary}}

Rate the fluency of the summary on a scale of 1 to 5 and explain your rating. Please use the format of: ##Rating: {rating} ##Explanation: {explanation}.
```

Figure 12: Prompt template used by Llama-3 (8B) model as a judge to evaluate the fluency of a RAG response. We first explain the criteria for evaluation and the model outputs an explanation and score between [1,5] indicating the fluency of the output. {{language}} is a placeholder for the target language; {{Question}} is a placeholder for the MIRAGE-BENCH query; {{Documents}} is a placeholder for both MIRAGE-BENCH relevant and non-relevant passages concatenated together; {{Summary}} is a placeholder for RAG model output.

Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants tasked to answer the question displayed below, based on a set of documents retrieved by a search engine.

You should choose the assistant that best answers the user question based on a set of reference documents that may or not be relevant referenced in the IEEE format.

Your evaluation should consider factors such as the correctness, helpfulness, completeness, accuracy, depth, and level of detail of their responses.

Details are only useful if they answer the user's question. If an answer contains non-relevant details, it should not be preferred over one that only uses relevant information.

Begin your evaluation by explaining why each answer correctly answers the user question. Then, you should compare the two responses and provide a short explanation on their differences. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Be as objective as possible.

After providing your explanation, output your final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]" if assistant B is better, and "[[C]]" for a tie.

```
"[User Question]"
{{query}}

"[Reference Documents]"
{{documents}}

"[The Start of Assistant A's Answer]"
{{answer_a}}

"[The End of Assistant A's Answer]"

"[The Start of Assistant B's Answer]"
{{answer_b}}

"[The End of Assistant B's Answer]"
```

Figure 13: Prompt template used by LLM as a judge to evaluate the RAG response in a pairwise evaluation involving a head-to-head battle. The template is taken and modified from RAGEval (Rackauckas et al., 2024). We explain the evaluation criteria and ask the judge to evaluate two RAG responses based on multiple factors, including correctness, helpfulness, completeness, accuracy, depth, and level of detail. The Judge provides a justification for their model choice and at the end of the response indicates as either "[[A]]", "[[B]]", or "[[C]]" denoting a tie. {{query}} is a placeholder for the input MIRAGE-BENCH query; {{documents}} is a placeholder for both MIRAGE-BENCH relevant and non-relevant passages concatenated together; {{answer\_a}} is a placeholder for the output response of model A; {{answer\_b}} is a placeholder for the output response of model B.