2009–2012) [2] and the filtered version of ClariQ proposed by Sekulic et al. [27], which maps clarifying questions to facets. For QA, we use PaQa (AmbigNQ with GPT-3 clarifications) [8] and CambigNQ (AmbigNQ queries augmented with human-validated clarifications) [11].

Adapting Datasets for Retrieval-Augmented Clarification. Existing clarification datasets (Qulac, ClariQ) lack passage-level grounding, as their relevance labels are assigned at the document level and not explicitly tied to the clarifying question. To bridge this gap, we derive passage-level supervision through a three-stage pipeline: (i) *Passage Indexing*: we segment Clueweb09-12[4] into 250-token passages, following TREC CAsT [20], and index them with Pyserini [15]; (ii) *Query Rewriting*: for each ambiguous query–clarification pair (U_q, C_q), we generate a facet-specific reformulation U_q^r by incorporating C_q using an LLM, yielding sharper retrieval intents than U_q alone; (iii) *Pseudo-Relevance Retrieval*: we employ BM25 [25] over the passage index to retrieve the top-k passages $\mathcal{D}$ for U_q^r, treating them as pseudo-relevant evidence. This produces training tuples $(U_q, \mathcal{D}, C_q)$ that support retrieval-conditioned clarification generation.

Metrics. We employ both reference-based and reference-free metrics to evaluate the quality of generated clarifying questions. Reference-based metrics measure similarity to gold questions, while reference-free metrics assess faithfulness to the input query and associated passages. In addition, we use GPT-4 to assess faithfulness, serving as a model-based proxy for human judgment.

Reference-based evaluation. We report BLEU [21], ROUGE-L [14], METEOR [3], and BERTScore [33]. BLEU and ROUGE-L capture n-gram and longest common subsequence overlap, respectively, while METEOR accounts for synonym and stem matches. BERTScore computes semantic similarity via contextualized token embeddings, providing a finer-grained assessment of meaning preservation. These metrics are consistent with prior work in clarification question generation and facilitate direct comparison.

Faithfulness evaluation. We evaluate faithfulness using PARENT Recall (PAR) [5] and AlignScore (AL) [13]. PAR, originally proposed for data-to-text generation, computes n-gram recall against both the input and the reference, serving as a proxy for input-groundedness. To apply it to unstructured passages, we adapt the metric by extracting named entities, multi-word noun phrases, and subject–verb–object triples with SpaCy[5], allowing content-level overlap measurement without reliance on structured data. AL is an entailment-based metric built on RoBERTa [16] and trained on multiple NLI datasets. Because clarifying questions are often interrogative and not well-suited for direct entailment evaluation, we convert them into declarative statements by removing question templates, retaining only content-bearing tokens, and filtering query overlaps.

[4] https://lemurproject.org/clueweb09/
[5] https://spacy.io/

This yields hypotheses compatible with AL's premise–hypothesis structure while preserving the semantic content of the questions.

4.2 Baselines

We evaluate RAC against several baselines. First, we include (`AT-CoT`), the ambiguity taxonomy chain-of-thought prompting baseline of Tang et al. [29], which applies few-shot prompting conditioned only on the query. Following Sekulic et al. [27], we use the widely adopted (`Q-Cond`) fine-tuned model, which generates clarifications from the query alone. To assess the impact of supervision, we compare RAC to a ($\texttt{QP-Zero}_{\texttt{shot}}$) variant conditioned on both query and passages in a zero-shot setting. Finally, on ClariQ, where facet annotations are available, we also report results for the template-based (`TB`) and facet-based (`QF-Cond`) baselines of Sekulic et al. [27]. For LLM-based methods, we use the same underlying model to ensure a fair comparison.

4.3 Implementation Details and Hyperparameters

We build on the pre-trained `LLaMA3.1-8B-base` checkpoint from the HuggingFace Hub, using the `Transformers` and `TRL` libraries [31]. For supervised fine-tuning (SFT), we train for 2 epochs with a learning rate of 1×10^{-5}, batch size 32, and a linear learning rate schedule. For direct preference optimization (DPO), we use 2 epochs with a learning rate of 2×10^{-6}, batch size 32, and $\beta = 0.1$. In our joint loss, we set $\gamma = 0.5$, based on ablation results. Zero-shot baselines rely on the `Instruct` variant of the base model. All experiments are run on NVIDIA A100 GPUs (80GB). Source code is available at: `https://github.com/RayaneA7/RAC-Retrieval-augmented-clarifcation`.

5 Results

5.1 Main Results

The main evaluation results are reported in Table 1. We find that *RAC* significantly outperforms the baselines across all metrics and datasets, confirming that passage conditioning substantially improves clarifying question generation, answering **RQ2**.

Moreover, results show that reference-based measures fail to capture the gains from preference tuning, consistent with prior findings [4,6,23]. In contrast, reference-free evaluation –reported only for models conditionned with passages– reveals that RAC_{DPO} achieves better performance over RAC_{SFT}. This demonstrates that preference-based optmization enhances corpus faithfulness beyond sepervised fine-tuning, directly adressing **RQ3**.

The fact that QP-Zero performs significantly worse than Q-cond highlights the importance of learning the form of a clarification question, independently of its content.

Table 1: Evaluation scores of RAC variants against different baselines, with $\beta = 0.1$ and for mixture $\alpha = 0.7$. Bold values indicate best performance, and † indicates a statistically significant improvement (Welch's t-test, $p < 0.001$).

Dataset	Model	ROUGE-L ↑	BLEU ↑	METEOR ↑	BERTScore (F1) ↑	ALScore ↑	Par-R ↑
Conversational Search Datasets							
Qulac	AT-CoT	17.97	2.77	20.81	84.72	–	–
	Q-Cond	29.44	10.51	25.92	88.24	–	–
	QP-Zero$_{\text{shot}}$	27.39	5.68	33.33	87.20	–	–
	RAC$_{\text{SFT}}$(ours)	**33.14**†	**12.59**†	**31.30**†	**89.34**†	79.14	42.53
	+ **RAC$_{\text{DPO}}$(ours)**	**32.42**†	**11.52**†	**31.48**†	**88.92**†	**81.73**	**44.83**
ClariQ	AT-CoT	18.63	3.49	21.19	84.74	–	–
	Q-Cond	28.68	11.19	25.47	88.16	–	–
	TB	35.50	0.28	24.26	87.65	–	–
	QF-Cond	33.70	2.20	**37.56**	89.08	–	–
	QP-Zero$_{\text{shot}}$	26.03	4.99	31.81	86.59	–	–
	RAC$_{\text{SFT}}$(ours)	**36.25**†	**14.88**†	**34.01**†	**89.52**†	51.32	53.15
	+ **RAC$_{\text{DPO}}$(ours)**	**35.52**†	**14.86**†	**33.84**†	**89.39**†	**52.41**	**55.77**
Question Answering Datasets							
PaQa	AT-CoT	23.59	7.07	22.93	85.97	–	–
	Q-Cond	42.46	16.62	41.58	90.12	–	–
	QP-Zero$_{\text{shot}}$	33.79	10.42	35.84	88.66	–	–
	RAC$_{\text{SFT}}$(ours)	**46.83**†	**20.17**†	**47.97**†	**90.85**†	43.36	27.62
	+ **RAC$_{\text{DPO}}$(ours)**	**45.26**†	**18.32**†	**46.40**†	**90.41**†	**45.75**	**28.54**
CAmbigNQ	AT-CoT	10.33	2.10	8.53	84.02	–	–
	Q-Cond	28.41	8.90	33.06	87.17	–	–
	QP-Zero$_{\text{shot}}$	18.20	4.27	19.48	85.15	–	–
	RAC$_{\text{SFT}}$(ours)	**36.66**†	**14.81**†	**43.37**†	**88.93**†	47.62	87.99
	+ **RAC$_{\text{DPO}}$(ours)**	**35.47**†	**14.40**†	**41.99**†	**88.89**†	**49.95**	**88.05**

These findings highlight both the benefit of passage conditioning and the added value of preference-based optimization. We further validate these results through qualitative analysis and LLM-based judgments in subsequent experiments.

5.2 LLM-based Evaluation

To further address **RQ2**, we evaluate the faithfulness of our approach using GPT-4 as a evaluator, comparing RAC_{DPO} against RAC_{SFT}. Results are shown in Table 2. Across all datasets, RAC_{DPO} achieves higher win rates compared to RAC_{SFT}, in some cases by more than a factor of two, whereas a large fraction of outputs are judged as ties. These results suggest that supervised fine-tuning already provides a strong baseline, preference optimization yield further gains on harder cases, reinforcing **RQ3** by enhancing faithfulness beyond supervised training.

5.3 Impact of the Number of Input Passages

We next examine the impact of the number and quality of retrieved passages on RAC. Because RAC relies on retrieval to expose potential ambiguities, both the quantity and relevance of the input passages directly affect its ability to generate effective clarifications.