

Intent-Driven Generative Query Suggestion with Query-Level Credit Assignment

Xinpeng Liu^{1,*} Lu Ma^{2,*} Jiayi Qiao³ Mengyu Zhou^{2,†} Linglong Li^{4,†}

Xiaofeng Bian² Haonan Chen² Xiaoxi Jiang² Guanjun Jiang²

¹School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University

²Qwen Large Model Application Team, Alibaba

³National University of Singapore

⁴Pengcheng National Laboratory

Abstract

Generative query suggestion aims to enhance user engagement by anticipating user intents and recommending relevant follow-up queries. A central challenge is to generate slates whose individual queries are useful while the slate covers distinct intents. We propose an Intent-Driven Query Suggestion Framework with dual-stage optimization. First, *intent-aware diversity modeling* constructs intent-aligned supervised fine-tuning (SFT) data and uses an Intent-Aware Diversity Reward to optimize intent coverage. Second, *query-level credit assignment* routes individual quality signals to the corresponding query tokens while sharing a slate-level diversity signal across the slate. Experiments on a large-scale production dataset, including online A/B testing and offline evaluation, show improvements in click-through rate, query quality, and intent coverage.

1 Introduction

The rise of conversational AI assistants (Achiam et al., 2023) has fundamentally transformed how users access information and accomplish tasks. Recent work studies multimodal, click-guided, and preference-aligned query suggestion for conversational systems (Wang et al., 2024; Min et al., 2025; Yin et al., 2025). Figure 1 shows an example of conversational query suggestion, where diverse follow-up queries are recommended to enhance user engagement.

Recent advances (Lin et al., 2023; Bao et al., 2023) in generative query suggestion employ large language models (LLMs) to synthesize queries directly from conversational context. While gen-

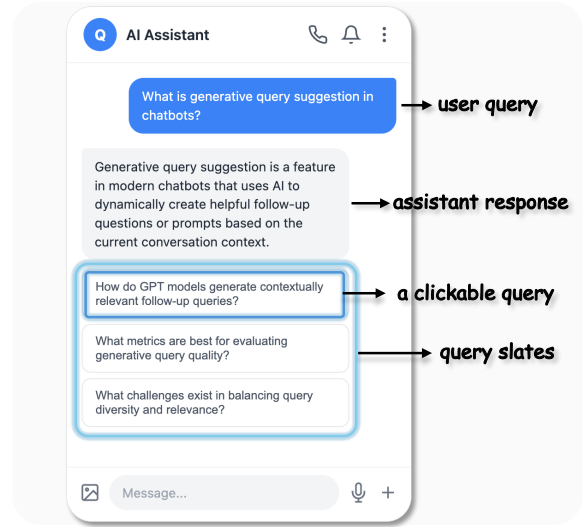


Figure 1: Example of conversational query suggestion, where the AI Assistant recommends diverse follow-up queries to enhance user engagement.

erative query suggestion offers greater flexibility than retrieval-based methods, it requires dual optimization: query slates must be diverse to cover the breadth of potential user follow-up interests, while each individual query must remain contextually relevant to the conversation history, accurately capturing the user’s underlying intent.

However, jointly optimizing these objectives presents two challenges. **L1) Intent-aware diversity optimization.** Semantic diversification through embedding similarity and post-hoc reranking is well established, but such methods do not necessarily optimize context-dependent coverage of distinct user intents during policy training. For instance, “How to learn Python” and “Python tutorials” are lexically distinct but express the same learning intent. We therefore model diversity as coverage over orthogonal intent dimensions rather than textual variation alone. **L2) Within-slate credit**

*Equal contribution.

†Corresponding authors.

This work was supported by the National Science and Technology Major Project (2024ZD01NL00101).

assignment. Min et al. (2025) and Yin et al. (2025) align query suggestions with click- or preference-based objectives. Our method instead explicitly applies query-specific quality advantages only to the corresponding query tokens while sharing a slate-level intent-coverage advantage across the slate. This decomposition identifies each query’s contribution within an autoregressively generated slate.

To address these challenges, we propose an **Intent-Driven Query Suggestion Framework** with dual-stage optimization for conversational AI assistants. To address L1, we introduce **intent-aware diversity modeling** in semantic intent space. During SFT, we construct a cold-start data pipeline based on user-initiative alignment. During reinforcement learning (RL), an Intent-Aware Diversity Reward (IAD-R) captures intent coverage through Noisy-OR aggregation. To address L2, we devise a **query-level credit assignment mechanism**: query-quality advantages are applied only to the corresponding query tokens, whereas a slate-level diversity advantage is shared across all queries. Their composite enables fine-grained optimization of individual quality while maintaining overall slate diversity.

We validate the framework through industrial online A/B testing and human evaluation. Its distinguishing feature is the combination of query-specific token-level advantages and a shared slate-level intent-coverage advantage in a single policy update.

2 Related Work

Generative Query Suggestion Large Language Models (LLMs) have been widely used to generate diverse and context-aware query suggestions (Di Palma, 2023). Diversification can be imposed through decoding constraints (Deng et al., 2025; Guo et al., 2025), embedding-based reranking (Bacciu et al., 2024), or preference alignment. Min et al. (2025) combine multi-source CTR modeling, diversity-aware CTR-weighted DPO, and iterative CTR calibration; Yin et al. (2025) introduce uncertainty-aware preference modeling, out-of-distribution regularization, and multi-stage alignment. These systems align suggestions with clicks or preferences. Our focus instead is the granularity of credit within a generated slate: query-specific quality advantages are routed only to the corresponding tokens, while intent coverage supplies a shared slate-level advantage. Inspired by

exposure modeling in recommender systems (Yin et al., 2025; Min et al., 2025; Chen et al., 2025), we view each generated query as activating a user intent and optimize intent-space coverage directly.

Credit Assignment and Slate Optimization in RL Reinforcement Learning (RL) is commonly used to align generative models with human preferences (Rafailov et al., 2023; Ouyang et al., 2022), but slate-level rewards often lead to ambiguous credit assignment (Yin et al., 2025). Prior studies address this with process reward modeling at the token or step level (Yue et al., 2025; Xiong et al., 2025; Cui et al., 2025). SlateQ (Ie et al., 2019) decomposes slate Q-values into item-level scores over a fixed candidate pool, while our setting constructs slates autoregressively and optimizes them with policy gradients. We therefore introduce query-level credit assignment, which distributes slate rewards into per-query advantages and combines them with intent-aware diversity rewards.

3 Methodology

We formalize conversational query suggestion as a slate generation task (§3.1) that jointly optimizes query-level quality and slate-level diversity. As illustrated in Figure 2, our framework consists of two stages: (1) a cold-start supervised fine-tuning stage (§3.2) that constructs high-quality training data through LLM generation and human-AI collaboration, and (2) a reinforcement learning stage (§3.3) that optimizes intent-aware diversity and query-level credit assignment.

3.1 Problem Formulation

We formulate conversational query suggestion as a conditional generation task. Given context $\mathcal{X} = \{q_u, r, h\}$, where q_u denotes the current user query, r the assistant response, and h the conversation history, the model generates a slate of K candidate queries:

$$\mathcal{S} = \{q_1, \dots, q_K\} \sim \pi_\theta(\cdot | \mathcal{X}) \quad (1)$$

where π_θ represents the policy model parameterized by θ . The objective is to jointly optimize individual query quality and slate-level diversity.

3.2 Cold-Start SFT

To address the data quality and diversity bottleneck in the SFT, we design a two-stage data construction pipeline coupled with a hybrid training strategy.

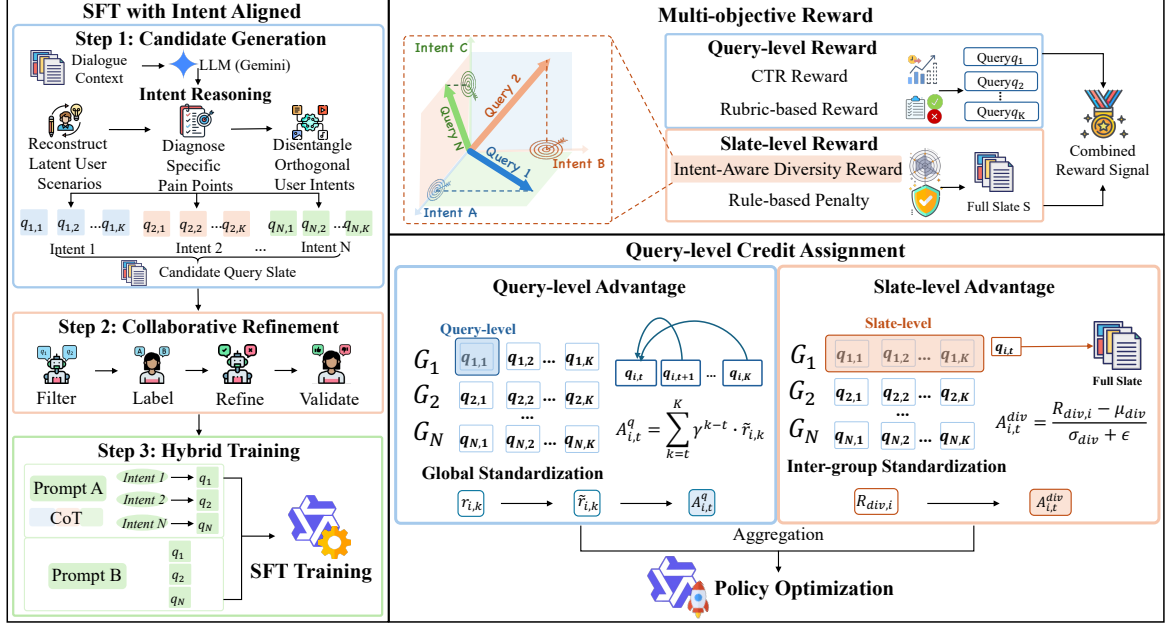


Figure 2: Overview of our framework. Left: Cold-Start SFT stage with LLM generation, human-AI collaboration and hybrid CoT training. Right: RL stage with multi-objective reward and query-level credit assignment.

Step 1: Candidate Generation. We leverage LLMs to perform multi-dimensional intent analysis on dialogue context, decomposing user needs into orthogonal intent dimensions through scenario reconstruction and pain point diagnosis. For each identified intent, we generate a diverse set of candidate queries that address distinct user information needs.

Step 2: Collaborative Refinement. We adopt a multi-stage human-AI collaboration pipeline to ensure data quality. The LLM first filters the optimal query for each intent dimension. Human annotators and an LLM-based evaluator then iteratively label and cross-validate the samples, with disagreements surfaced for re-adjudication to reduce subjective variability. This process ensures both scalability and high annotation quality. Detailed annotation protocols, qualification criteria, and quality control procedures are provided in Appendix C.

Step 3: Hybrid Training. To balance inference efficiency with reasoning capability, we construct a hybrid instruction dataset containing both Direct Query Generation and Intent-Aware Chain-of-Thought (CoT) samples. Each generation mode is guided by task-specific system prompts (representative excerpts appear in Appendix Figures 4 and 5). This hybrid approach enables the model to capture complex intent reasoning patterns while maintaining real-time generation throughput.

3.3 Reinforcement Learning

3.3.1 Multi-Objective Reward

CTR Reward To align suggestions with user preferences, we employ a lightweight BERT-based CTR model trained on historical click logs. Given conversational context $\mathcal{X}^{(\tau)} = \{q_u^{(\tau)}, r^{(\tau)}, h^{(\tau)}\}$ and candidate query q_i , the model predicts click probability:

$$R_{CTR}(q_i, \mathcal{X}^{(\tau)}) = \hat{p}(q_i | \mathcal{X}^{(\tau)}) \in [0, 1] \quad (2)$$

This reward guides the policy toward generating user-preferred suggestions.

Rubric-based Reward To prevent reward models from exploiting surface-level patterns, we introduce *Rubric-based Rewards* following (Ye et al., 2023; Liu et al., 2023). We employ an LLM to assess whether each query satisfies expert-curated quality rubrics (summarized in Appendix Figure 6). Given query q and rubrics $\mathcal{R} = \{r_1, \dots, r_n\}$, the reward uses an all-pass criterion:

$$R_{rubrics}(q) = \begin{cases} 1, & \text{if } \forall i, r_i(q) = 1 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where $r_i(q) \in \{0, 1\}$ indicates whether query q satisfies the i -th rubric. During RL training, Qwen-Max produces this rubric-based reward using the rubric structure summarized in Appendix Figure 6.

Rule-based Penalty We impose rule-based constraints $R_{\text{rule}}(\hat{\mathcal{Q}}_{(\tau)}^{(t)})$ to ensure quality and prevent degenerate outputs: (1) redundancy penalty based on Jaccard similarity, (2) length penalty, and (3) format compliance penalty.

Intent-Aware Diversity Reward In deployed generative query suggestion, commonly used surface deduplication and post-hoc reranking do not explicitly optimize context-dependent intent coverage during training.

Given conversation history $h^{(\tau)}$, we leverage an LLM to infer M orthogonal user intents $\mathcal{Z} = \{z_1, \dots, z_M\}$ during offline construction, verified by human annotators. We adopt a Noisy-OR model (Pearl, 1988) to measure slate $\mathcal{S}$'s coverage over the intent space. For intent z_m , the coverage probability represents the likelihood that at least one query in the slate activates this intent:

$$P_{\text{cov}}(z_m | \mathcal{S}) = 1 - \prod_{q_k \in \mathcal{S}} (1 - \sigma(\text{sim}(q_k, z_m))) \quad (4)$$

where $\text{sim}(q_k, z_m) = \frac{\phi(q_k)^\top \phi(z_m)}{\|\phi(q_k)\|_2 \|\phi(z_m)\|_2}$ is the cosine similarity between query and intent embeddings from Sentence-BERT (Reimers and Gurevych, 2019), and $\sigma(\cdot)$ is the sigmoid function. The Intent-Aware Diversity Reward (IAD-R) averages coverage across all intents:

$$R_{\text{div}}(\mathcal{S}) = \frac{1}{M} \sum_{m=1}^M P_{\text{cov}}(z_m | \mathcal{S}) \quad (5)$$

We employ uniform weighting to ensure unbiased intent coverage. This prevents the model from prioritizing dominant intents at the expense of long-tail scenarios.

3.3.2 Query-level Credit Assignment

Our query-level credit assignment mechanism includes query-level quality advantages and slate-level diversity advantages.

Query-level Advantage Calculation For G query slates $\{y_i\}_{i=1}^G$ sampled from policy π_θ , where each slate consists of K queries, we compute the query-level reward for the k -th query in the i -th slate as a weighted combination of rubric-based and CTR rewards:

$$r_{i,k} = \alpha \cdot \mathcal{R}_{\text{rubrics}}(q_{i,k}) + \beta \cdot \mathcal{R}_{\text{CTR}}(q_{i,k}) \quad (6)$$

Here, $\alpha = 0.6$ weights rubric compliance and $\beta = 0.4$ weights predicted CTR. The aggregated query-level reward ensures that the generated queries

maintain semantic compliance and quality stability while maximizing actual click-through conversion efficiency.

In contrast to GRPO's group-level normalization over sequence returns, we perform query-level advantage estimation by globally normalizing rewards across all $G \times K$ queries:

$$\tilde{r}_{i,k} = \frac{r_{i,k} - \mu_r}{\sigma_r + \epsilon} \quad (7)$$

where μ_r and σ_r denote the mean and standard deviation computed over all sampled query rewards. To account for the sequential dependency between queries, we introduce a discount factor $\gamma \in [0, 1]$ and compute the advantage for the t -th query as:

$$A_{i,t}^q = \sum_{k=t}^K \gamma^{k-t} \cdot \tilde{r}_{i,k} \quad (8)$$

This formulation values each query by both its immediate reward and its impact on subsequent generations, encouraging coherent and high-quality query sequences.

Slate-level Diversity Advantage To encourage diverse query lists, we introduce slate-level diversity advantages. As diversity is a collective property, we normalize the diversity reward $R_{\text{div},i}$ within the group:

$$A_{i,t}^{\text{div}} = \frac{R_{\text{div},i} - \mu_{\text{div}}}{\sigma_{\text{div}} + \epsilon}, \quad \forall t \in \{1, \dots, K\} \quad (9)$$

where $\mu_{\text{div}}, \sigma_{\text{div}}$ are computed over $\{R_{\text{div},j}\}_{j \in G}$. This advantage is uniformly assigned to all queries, encouraging complementary suggestions.

Advantage Aggregation and Policy Optimization We combine query quality and slate diversity through a composite advantage with rule-based gating:

$$A_{i,t} = \begin{cases} \frac{A_{i,t}^q + A_{i,t}^{\text{div}}}{\sqrt{2}}, & \text{if Rule}(y_i) = \text{Pass} \\ -1, & \text{if Rule}(y_i) = \text{Fail} \end{cases} \quad (10)$$

where quality and diversity advantages are equally weighted and normalized. Slates violating critical constraints receive -1 penalty, combining learned and rule-based rewards.

Following GRPO, we optimize token-level generation probabilities using query-level advantages. To bridge this granularity gap, we uniformly broadcast each query-level advantage to all its tokens:

$$A_j^{\text{tok}} = A_{i,t}, \quad \forall u_j \in \text{tokens}(q_{i,t}) \quad (11)$$

Method	CTR Gain (%) $\uparrow$	GSB vs. base $\uparrow$	LLM Critic $\uparrow$	Self-BLEU $\downarrow$	Intent Cov. $\uparrow$
PE-based	+0	+0	0.33	0.0656	0.27
SFT-cold start	+27.06	+0.16	0.67	0.0609	0.54
PPO	+35.29	+0.31	0.71	0.0571	0.69
GRPO	+43.53	+0.37	0.83	0.0474	0.85
RL-Ours	+48.24	+0.44	0.88	0.0470	0.91

Table 1: Main results. CTR reports relative gains over the PE-based arm; GSB is relative to PE-based outputs. Bold indicates the best result.

The policy is then optimized via the clipped surrogate objective:

$$\mathcal{L}(\theta) = -\mathbb{E}_{y \sim \pi_{\theta}(\cdot|x)} \left[\sum_j \min(\rho_j A_j^{\text{tok}}, \bar{\rho}_j A_j^{\text{tok}}) - \lambda_{\text{KL}} D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}}) \right]. \quad (12)$$

where $\rho_j = \pi_{\theta}(u_j|u_{<j})/\pi_{\text{ref}}(u_j|u_{<j})$ is the probability ratio, $\bar{\rho}_j = \text{clip}(\rho_j, 1 - \epsilon, 1 + \epsilon)$, and λ_{KL} is the KL coefficient. The clipping mechanism prevents large policy updates, while the KL penalty ensures training stability.

4 Experiments

4.1 Experimental Setup

We evaluate on a large-scale production system using anonymized real-world interactions. We use Qwen3-30B-A3B (Yang et al., 2025) as the foundation model (training details in Appendix D).

Training Datasets. We collect data from real user logs of a large-scale conversational AI assistant. For SFT, we construct 15,000 instances from 4 weeks of conversations using the pipeline in §3.2, mixing 20% intent-augmented CoT samples with 80% direct generation samples. For RL, we sample 12,000 instances from an additional 2 weeks, annotated with intents for IAD-R.

Test Datasets. We construct two expert-validated test sets spanning diverse user scenarios. The *Offline Evaluation Set* contains 500 contexts for automatic metrics. The *Human Evaluation Set* contains 150 representative contexts for measuring generation quality and diversity.

Online Metrics. To assess real-world efficacy, we conducted a one-week A/B test on live traffic. The core metric is *Click-Through Rate* (CTR), which measures interaction with the suggestion slate within a single turn. The five systems ran concurrently in mutually exclusive, randomly assigned user-level buckets. Traffic allocation for PE-based/SFT/PPO/GRPO/RL-Ours was 5/80/5/5/5%;

SFT was the incumbent production arm, while Table 1 reports CTR gains relative to the PE-based arm. We use two-proportion z -tests on unrounded arm-level CTRs. Exact traffic counts and absolute CTR values are commercially sensitive.

Offline Metrics. We evaluate query quality with: (1) *Good-Same-Bad* (GSB), a pairwise side-by-side human evaluation across content quality, text quality, intent relevance, and information gain (Table 9), yielding $(\# \text{Good} - \# \text{Bad})/N$; and (2) *LLM Critic*, an ensemble of Gemini-3, Claude-4.5, and Qwen-Max that reports the majority-voted checklist pass rate (Figure 6). Thus, no single Qwen-family evaluator determines this metric. For slate diversity, we use: (1) *Self-BLEU* (Yin et al., 2025; Min et al., 2025), the average pairwise similarity within each slate; and (2) *Intent Coverage*, the recall of separately annotated test intents, where three annotators independently label intent matches for each query and majority voting determines the final label.

Intent Coverage targets the same property as IAD-R, and LLM Critic uses the same rubric family as the training reward. We therefore treat both as target-aligned diagnostics rather than fully independent evidence. Neither evaluation directly reuses stored training-time reward outputs: evaluation coverage uses discrete human matching against separately annotated test intents instead of Sentence-BERT similarity and Noisy-OR aggregation, while LLM Critic recomputes judgments on held-out contexts using the three-model ensemble. Because Qwen-Max participates in both training and the evaluation ensemble, online CTR and human GSB serve as the primary independent evidence.

4.2 Overall Results and Analysis

Table 1 compares our RL framework with the production and policy-optimization baselines. Scalability to Qwen3-235B-A22B is evaluated in Appendix B.

SFT Establishes a Strong Foundation. Starting from the PE-based prompt engineering baseline, supervised fine-tuning on user interaction data

Method	LLM Critic $\uparrow$	Self-BLEU $\downarrow$	Intent Cov. $\uparrow$
DPO	0.63	0.0723	0.55
CTR-weighted DPO	0.69	0.0714	0.49
GRPO	0.83	0.0474	0.85
RL-Ours	0.88	0.0470	0.91

Table 2: Controlled offline optimization comparison. The DPO variants use the same SFT initialization, training data, slate size, decoding settings, and evaluation budget. CTR-weighted DPO is inspired by [Min et al. \(2025\)](#), not a full reproduction.

improves all five metrics. As shown in Table 1, the SFT-cold start strategy achieves a +27.06% CTR improvement, a GSB score of +0.16, and an LLM Critic score of 0.67. These results support the value of learning from real user interactions beyond prompt engineering.

RL Methods Improve Performance. We explore PPO and GRPO using the same reward signals as our method. PPO increases the PE-relative CTR gain to +35.29%. GRPO reaches +43.53% CTR gain, +0.37 GSB, and 0.83 LLM Critic. Relative to the incumbent SFT arm, GRPO improves CTR by 13.0% ($p = 0.006$).

Query-level Credit Assignment Achieves the Best Results. Our method attains the highest PE-relative CTR gain (+48.24%). The 4.71 difference from GRPO is measured in percentage points of PE-relative gain; the direct comparison is a 3.25% relative CTR improvement ($p = 0.02$). RL-Ours also improves CTR over the incumbent SFT arm by 16.7% ($p = 0.003$). On the common 150-context paired evaluation subset, two-sided paired t -tests comparing RL-Ours with GRPO give GSB +0.07 ($p = 0.03$), Intent Coverage +0.06 ($p = 0.002$), and LLM Critic +0.05 ($p = 0.05$). Self-BLEU changes from 0.0474 to 0.0470, whereas Intent Coverage increases from 0.85 to 0.91. Because Intent Coverage and LLM Critic are aligned with their corresponding training objectives, CTR and human GSB are the primary independent evidence for the improvement.

Table 2 adds standard DPO ([Rafailov et al., 2023](#)) and a CTR-weighted DPO baseline. Both underperform GRPO and RL-Ours on the three available offline metrics. The CTR-weighted variant improves LLM Critic and Self-BLEU over standard DPO but reduces Intent Coverage, suggesting that pairwise click-weighted optimization does not directly capture slate-level intent coverage. We do not report online CTR or human GSB for these additional runs. [Yin et al. \(2025\)](#) cannot be repro-

Method	GSB vs. base	LLM Critic	Self-BLEU	Intent Cov.
PE-based	+0	0.33	0.0656	0.27
SFT-Candidate	+0.05	0.41	0.0631	0.28
SFT-Refined	+0.13	0.59	0.0617	0.42
SFT-Hybrid	+0.16	0.67	0.0609	0.54

Table 3: Ablation study on data pipeline stages. Self-BLEU $\downarrow$, other metrics $\uparrow$.

Method	Δ GSB vs. RL-Ours	LLM Critic	Self-BLEU	Intent Cov.
RL-Ours	0.00	0.88	0.0470	0.91
w/o IAD-R	-0.12	0.83	0.0461	0.71
w/o Rubric	-0.09	0.62	0.0496	0.86
w/o CTR	-0.11	0.76	0.0486	0.89
w/o query-level	-0.07	0.83	0.0474	0.85

Table 4: Ablation on RL components. GSB is measured relative to RL-Ours; the other columns report absolute values. “query-level” refers to query-level credit assignment. Self-BLEU $\downarrow$, other metrics $\uparrow$.

duced in full because its reward models, data, and training components are unavailable; the conceptual comparison appears in Section 2.

4.3 Ablation Studies

Ablation Study on SFT Data. To validate our data construction pipeline (Section 3.2), we conduct an ablation study using different data sources: *PE-based* uses only prompt engineering without fine-tuning, *SFT-Candidate* trains on LLM-generated candidates (Stage 1), *SFT-Refined* trains on human-AI collaborative refined queries (Stage 2), and *SFT-Hybrid* trains on the full pipeline with mixed CoT traces and queries (Stage 3).

Table 3 shows progressive improvements across pipeline stages. Relative to SFT-Candidate, SFT-Refined improves GSB by 0.08 and Intent Coverage by 0.14, supporting the value of human-AI refinement. SFT-Hybrid adds a further 0.12 in Intent Coverage, suggesting that mixing CoT traces with direct queries helps the model learn intent-aware generation. We adopt SFT-Hybrid as the cold-start model for RL training.

Ablation Study on RL Components. We systematically remove key components from our RL framework to validate their individual contributions. Table 4 presents the final performance, while Figure 3 illustrates the training dynamics to provide deeper insights into how these components affect the learning process.

Removing diversity reward causes the largest Intent Coverage drop, decreasing from 0.91 to 0.71. Figure 3(a) shows models with this reward converge faster and to a higher plateau, reaching 0.91 by step 100, whereas the ablated variant continues to improve gradually, only approaching 0.71 by step 125. Both quality reward components prove

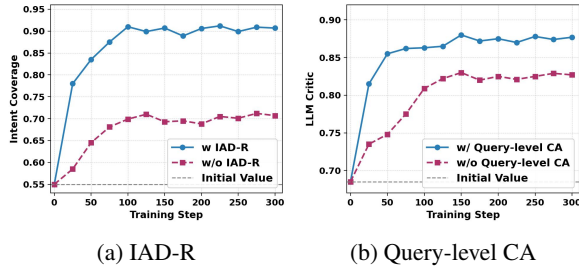


Figure 3: Training dynamics of ablated components. (a) IAD-R impact on Intent Coverage. (b) Query-level credit assignment on LLM Critic.

essential. Without rubric-based reward, the LLM Critic score drops from 0.88 to 0.62, showing its role in maintaining generation quality. Similarly, removing CTR reward decreases GSB by 0.11, suggesting that engagement signals effectively guide the model toward user-preferred queries.

Query-level credit assignment also outperforms the sequence-level baseline. Figure 3(b) demonstrates consistent advantages throughout training. As early as step 50, our method achieves an LLM Critic score of approximately 0.86 compared to 0.75 for the baseline, maintaining a consistent gap through convergence with final scores of 0.88 and 0.83 respectively. This result supports the conclusion that assigning rewards at the query level rather than the sequence level enables more effective optimization by providing precise feedback for each individual query.

5 Conclusion

We propose an intent-driven framework that jointly optimizes query quality and slate diversity through dual-stage optimization. The online and offline results show that query-level credit assignment improves engagement and human-evaluated quality while intent-aware training increases semantic coverage. Future work will explore personalization and multimodal query generation.

Limitations

This work has several limitations. First, our training and online evaluation are based on anonymized production logs from a deployed conversational assistant; although this setting provides realistic feedback, the proprietary nature of the data limits its direct reproducibility. We therefore report detailed data construction and evaluation protocols and plan to release a sanitized evaluation subset when compliance permits. Second, online CTR is an engagement proxy rather than a direct measure of user satisfaction or task success. We interpret it together with offline human evaluation, and future work should incorporate richer satisfaction and task-completion signals. The online analysis reports significance from a single one-week experiment but does not report confidence intervals or repeated-window variance. Third, Intent Coverage targets the same property as IAD-R, and LLM Critic uses the same rubric family as the training reward; Qwen-Max also participates in both training and the evaluation ensemble. The metrics do not reuse stored training-time reward outputs, but they are not fully independent evidence, so our main conclusions foreground online CTR and human GSB. Fourth, our evaluation uses independent annotations and majority voting but does not report a chance-corrected inter-annotator agreement statistic. Our intent-aware diversity reward also relies on LLM-assisted intent construction and human validation; remaining annotation noise or intent granularity mismatch may affect coverage estimation. Repeated full-scale RL runs and a complete sweep over the quality-diversity weights are computationally prohibitive, so we report evaluation-sample significance tests and only a targeted discount-factor sensitivity study. We also do not measure whether broader intent coverage improves multi-turn engagement conditioned on distinct follow-up intents. Finally, while query-level advantages improve optimization granularity, diversity remains a slate-level property shared across its queries; more precise marginal contribution estimation is an important direction for future work.

Ethics Statement

Use of AI Assistants We used ChatGPT solely as a writing assistant to polish the manuscript after the initial draft was written. The authors were responsible for all scientific content, analyses, and conclusions presented in this paper.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Andrea Bacciu, Enrico Palumbo, Andreas Damianou, Nicola Tonello, and Fabrizio Silvestri. 2024. Generating query recommendations via llms. *arXiv preprint arXiv:2405.19749*.
- Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2023. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. *arXiv preprint arXiv:2305.00447*.
- Jiahui Chen, Xiaozhe Jiang, Zhibo Wang, Quanzhi Zhu, Junyao Zhao, Feng Hu, Kang Pan, Ao Xie, Maohua Pei, Zhiheng Qin, and 1 others. 2025. Unisearch: Rethinking search system with a unified generative architecture. *arXiv preprint arXiv:2509.06887*.
- Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, and 1 others. 2025. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*.
- Jiabin Deng, Shiyao Wang, Kuo Cai, Lejian Ren, Qigen Hu, Weifeng Ding, Qiang Luo, and Guorui Zhou. 2025. Onerec: Unifying retrieve and rank with generative recommender and iterative preference alignment. *arXiv preprint arXiv:2502.18965*.
- Dario Di Palma. 2023. Retrieval-augmented recommender system: Enhancing recommender systems with large language models. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1369–1373.
- Xian Guo, Ben Chen, Siyuan Wang, Ying Yang, Chenyi Lei, Yuqing Ding, and Han Li. 2025. One-sug: The unified end-to-end generative framework for e-commerce query suggestion. *arXiv preprint arXiv:2506.06913*.
- Eugene Ie, Vihan Jain, Jing Wang, Sanmit Narvekar, Ritesh Agarwal, Rui Wu, Heng-Tze Cheng, Tushar Chandra, and Craig Boutilier. 2019. Slateq: A tractable decomposition for reinforcement learning with recommendation sets. In *IJCAI*, volume 19, pages 2592–2599.
- Jianghao Lin, Xinyi Dai, Yunjia Xi, Weiwen Liu, Bo Chen, Xiangyang Li, Chenxu Zhu, Huifeng Guo, Yong Yu, Ruiming Tang, and Weinan Zhang. 2023. How can recommender systems benefit from large language models: A survey. *arXiv preprint arXiv:2306.05817*.
- Yang Liu, Dan Iter, Yichong Xu, Shuhang Wang, Ruochen Xu, and Chenguang Zhu. 2023. [G-eval: Nlg evaluation using gpt-4 with better human alignment](#). *Preprint*, arXiv:2303.16634.
- Erxue Min, Hsiu-Yuan Huang, Xihong Yang, Min Yang, Xin Jia, Yunfang Wu, Hengyi Cai, Junfeng Wang, Shuaiqiang Wang, and Dawei Yin. 2025. Ctr-guided generative query suggestion in conversational search. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 2624–2634.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Judea Pearl. 1988. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-bert: Sentence embeddings using siamese bert-networks](#). *Preprint*, arXiv:1908.10084.
- Zheng Wang, Bingzheng Gan, and Wei Shi. 2024. Multimodal query suggestion with multi-agent reinforcement learning from human feedback. In *Proceedings of the ACM Web Conference 2024*, pages 1374–1385.
- Wei Xiong, Wenting Zhao, Weizhe Yuan, Olga Golovneva, Tong Zhang, Jason Weston, and Sainbayar Sukhbaatar. 2025. Stepwis: Stepwise generative judges for wiser reasoning. *arXiv preprint arXiv:2508.19229*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. 2023. Flask: Fine-grained language model evaluation based on alignment skill sets. *arXiv preprint arXiv:2307.10928*.
- Junhao Yin, Haolin Wang, Peng Bao, Ju Xu, and Yongliang Wang. 2025. From clicks to preference: A multi-stage alignment framework for generative query suggestion in conversational system. *arXiv preprint arXiv:2508.15811*.
- Chuhuai Yue, Chengqi Dong, Yinan Gao, Hang He, Jiajun Chai, Guojun Yin, and Wei Lin. 2025. Promoting efficient reasoning with verifiable stepwise reward. *arXiv preprint arXiv:2508.10293*.

A Prompt Templates

We provide abridged prompt excerpts that illustrate the formats used for direct query generation (Figure 4), intent-guided CoT reasoning (Figure 5), and rubric-based evaluation (Figure 6). Some operational rules are omitted for brevity.

System Prompt for Direct Query Generation				
# Role				
User Intent Insight Expert tasked with generating follow-up queries on behalf of users.				
# Workflow				
1. Intent Categorization				
• Creative: Content generation/refinement tasks (writing, coding, translation). • Non-Creative: Factual Q&A, information retrieval, troubleshooting.				
2. Strategy				
• Creative: Generate instructive queries (style, audience, tone optimization). • Non-Creative: Explore extended value and practical needs.				
# Constraints				
• Generate 3-5 diverse queries (no duplicates or highly similar ones). • Character limit: 20 characters or fewer (preferably 8-16). • Avoid repeating historical queries or answered content.				
Example:				
What scenarios suit deep learning?				
How does ML do feature engineering?				
What GPUs for training models?				

Figure 4: Abridged system prompt for direct query generation.

B Scalability to Larger Models

Method	GSB vs base	LLM Critic	Self-BLEU	Intent Cov.
PE-based	+0	0.57	0.0493	0.58
SFT-cold start	+0.14	0.79	0.0471	0.76
RL-Ours	+0.23	0.92	0.0463	0.94

Table 5: Scalability results on Qwen3-235B-A22B. Self-BLEU ↓, other metrics ↑.

To validate the generalizability of our framework across model scales, we conduct experiments on Qwen3-235B-A22B using offline metrics due to deployment constraints. As shown in Table 5, our approach achieves 0.92 LLM Critic and 0.94 Intent Coverage compared with the baseline’s 0.57 and 0.58. Despite the larger model’s stronger base-

System Prompt for Intent CoT Query Generation			
You are an expert in user intent understanding. First, conduct user needs analysis, then provide queries the user wants to ask .			
# Input Description			
• [Session History] <history> • [User Query] <query> • [Agent Answer] <response>			
# Output Format			
<pre>{ "deeperIntent": { "userSituation": "<user profile>", "motivations": "<latent intent>", "potentialIntents": ["<Intent 1>", ...] }, "response_summary": { "answeredInDetail": "<fully answered>", "answeredNotDetail": "<partially>", "notAnswered": "<not covered>" }, "recommend_query": { "query1": "<query>", // 20 chars "reason1": "<reason>", ... } }</pre>			
# recommend_query_rule			
1. Content Repetition Issue: Recommended queries must not obviously repeat content from the user query or AI-Chat response; 2. ... (Additional rules omitted for brevity)			

Figure 5: Abridged system prompt for intent-guided CoT query generation.

line (0.57 vs. 0.33 LLM Critic on Qwen3-30B-A3B), our framework retains improvements of 61% in LLM Critic and 62% in Intent Coverage. Self-BLEU decreases from 0.0493 to 0.0471 and 0.0463, showing that SFT data construction and RL optimization also reduce lexical redundancy at the larger model scale.

Parameter	Value	Parameter	Value
Optimizer	AdamW	LR Scheduler	Linear
Learning Rate	1e-5	Grad Clip	1.0
β_1	0.99	β_2	0.999
Batch Size	128	Max Tokens	8192

Table 6: Hyper-parameters for the SFT stage.

We use $\gamma = 0.95$ and fix $K = 3$ because the production interface displays three suggestions; neither value was tuned on the test set. The rubric/CTR mixture $(\alpha, \beta) = (0.6, 0.4)$ is fixed across experiments. Table 8 reports a targeted offline sensitivity analysis with all other settings fixed.

System Prompt for Rubric-based Evaluation	
# Role	Senior Dialogue Recommendation Auditor. Evaluate whether recommended queries meet strict quality validation rules.
# Input Definitions	Query Intent, Current Time, Session History User Query, Agent Answer, Recommended Query
# Evaluation Rules	R01 (Awkward Expression): Grammatical issues, improper collocations, punctuation errors. R02 (Language Inconsistency): Must match the language of current user query. R03 (Outdated Questions): No outdated content unless historical context requested. R04 (Creative Content): For creative writing, use declarative/imperative only. R05 (Irrelevant Needs): Avoid content unrelated to user query or agent answer. ... (Additional rules)
# Output Format	JSON structure with two fields: is_pass: Boolean indicating pass/fail status violations: Array of violation objects containing rule_id, rule_name, and reason

Figure 6: Abridged prompt for rubric-based quality assessment.

Both metrics remain within a narrow range. Intent Coverage requires per-model human annotation and is therefore reported only for the selected $\gamma = 0.95$ setting (0.91 in Table 1).

C Annotation Pipeline and Quality Control

Annotation Pipeline. The collaborative refinement in Step 2 follows a four-stage procedure: (i) the LLM filters the optimal query for each intent dimension; (ii) human annotators verify and label a seed subset of samples to establish quality standards; (iii) an LLM-based evaluator performs batch labeling guided by human feedback; (iv) human validators review and refine the LLM-labeled results.

Annotator Training and Qualification. All annotators follow standardized guidelines and undergo rigorous training. Qualification requires achieving >90% agreement in back-to-back validation against gold-standard samples before being

Parameter	Value	Parameter	Value
Optimizer	AdamW	LR Schedule	Linear
LR	2e-6	Grad Clip	1.0
β_1	0.99	β_2	0.999
Batch Size	64	Max Tokens	9216
λ_{KL}	0.05	Rollout N	8
WD	0.1	Warmup	2
Clip Ratio	[0.2, 0.28]	Temp.	1.0
Top- p	1.0	Epochs	2
Rubric wt. α	0.6	CTR wt. β	0.4
γ	0.95	Slate Size K	3

Table 7: Hyper-parameters for the RL stage.

γ	0.90	0.95	0.99	1.00
LLM Critic $\uparrow$	0.871	0.880	0.868	0.873
Self-BLEU $\downarrow$	0.0491	0.0470	0.0482	0.0479

Table 8: Offline sensitivity to the query-level discount factor γ .

assigned to production labeling.

Quality Control. An independent QC team audits annotation quality on sampled batches. In addition, we apply machine-assisted cross-validation between LLM and human labels: samples with disagreement are automatically flagged and routed for re-adjudication. This mechanism reduces variability in subjective decisions and provides a continuous feedback loop for guideline refinement.

D Training Configuration

Model training is conducted using a cluster of 256 NVIDIA H100 GPUs. The specific training parameters for each stage are summarized in Tables 6 and 7.

E Algorithm Details

The complete pseudocode of our query-level credit assignment mechanism is provided in Algorithm 1.

Algorithm 1 Query-level Credit Assignment

Require: Policy π_θ , ref. policy π_{ref} , input x
Require: Hyperparams: $G, K, \alpha, \beta, \gamma, \epsilon, \lambda_{\text{KL}}$
Ensure: Updated policy parameters θ

- 1: **Step 1: Sample Generation**
- 2: **for** $i = 1$ **to** G **do**
- 3: Sample slate $y_i = \{q_{i,1}, \dots, q_{i,K}\} \sim \pi_\theta(\cdot | x)$
- 4: **end for**
- 5: **Step 2: Query-level Reward**
- 6: Compute $r_{i,k} = \alpha \cdot \mathcal{R}_{\text{rubrics}}(q_{i,k}) + \beta \cdot \mathcal{R}_{\text{CTR}}(q_{i,k})$
- 7: **Step 3: Query-level Advantage**
- 8: Normalize: $\tilde{r}_{i,k} = \frac{r_{i,k} - \mu_r}{\sigma_r + \epsilon}$
- 9: Compute: $A_{i,t}^q = \sum_{k=t}^K \gamma^{k-t} \cdot \tilde{r}_{i,k}$
- 10: **Step 4: Diversity Advantage**
- 11: Compute $R_{\text{div},i} = \mathcal{R}_{\text{div}}(y_i)$
- 12: Normalize: $A_{i,t}^{\text{div}} = \frac{R_{\text{div},i} - \mu_{\text{div}}}{\sigma_{\text{div}} + \epsilon}$
- 13: **Step 5: Advantage Aggregation**
- 14: $A_{i,t} = \begin{cases} (A_{i,t}^q + A_{i,t}^{\text{div}})/\sqrt{2}, & \text{Pass} \\ -1, & \text{Fail} \end{cases}$
- 15: **Step 6: Token-level Broadcasting**
- 16: Assign $A_j^{\text{tok}} = A_{i,t}$ for all $u_j \in \text{tokens}(q_{i,t})$
- 17: **Step 7: Policy Optimization**
- 18: Compute $\rho_j = \frac{\pi_\theta(u_j | u_{<j})}{\pi_{\text{ref}}(u_j | u_{<j})}$
- 19: $\mathcal{J} = \sum_j \min(\rho_j A_j^{\text{tok}}, \bar{\rho}_j A_j^{\text{tok}}) - \lambda_{\text{KL}} D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})$
- 20: $\mathcal{L}(\theta) = -\mathbb{E}[\mathcal{J}]$
- 21: Update: $\theta \leftarrow \theta - \nabla_\theta \mathcal{L}(\theta)$
- 22: **return** Updated policy π_θ

tor. Table 9 presents the detailed evaluation criteria across four dimensions: Content Quality, Requirement Satisfaction, Text Quality, and Information Gain.

F Case study

Traditional query suggestion models focus solely on the current turn, providing surface-level recommendations while neglecting global dialogue context and users' implicit needs. In contrast, our model integrates dialogue history with the current QA trajectory to perform multi-dimensional intent modeling across multiple facets: **Reconstructing Latent User Scenarios, Diagnosing Specific Pain Points, and Disentangling Orthogonal User Intent**s.

As illustrated in Figure 7, the baseline captures only surface-level intent, while our model performs implicit need inference and intent-driven analysis. The generated suggestions maintain contextual continuity and precisely address users' latent intentions.

G Offline Evaluation Standard

To systematically assess the contextual quality of the generated query suggestions, we establish a manual offline evaluation standard. The scoring protocol operates on a per-case basis, where each test case consists of three query suggestions generated from the preceding context. The score assigned to each case directly corresponds to the number of suggestions deemed valuable by the annota-

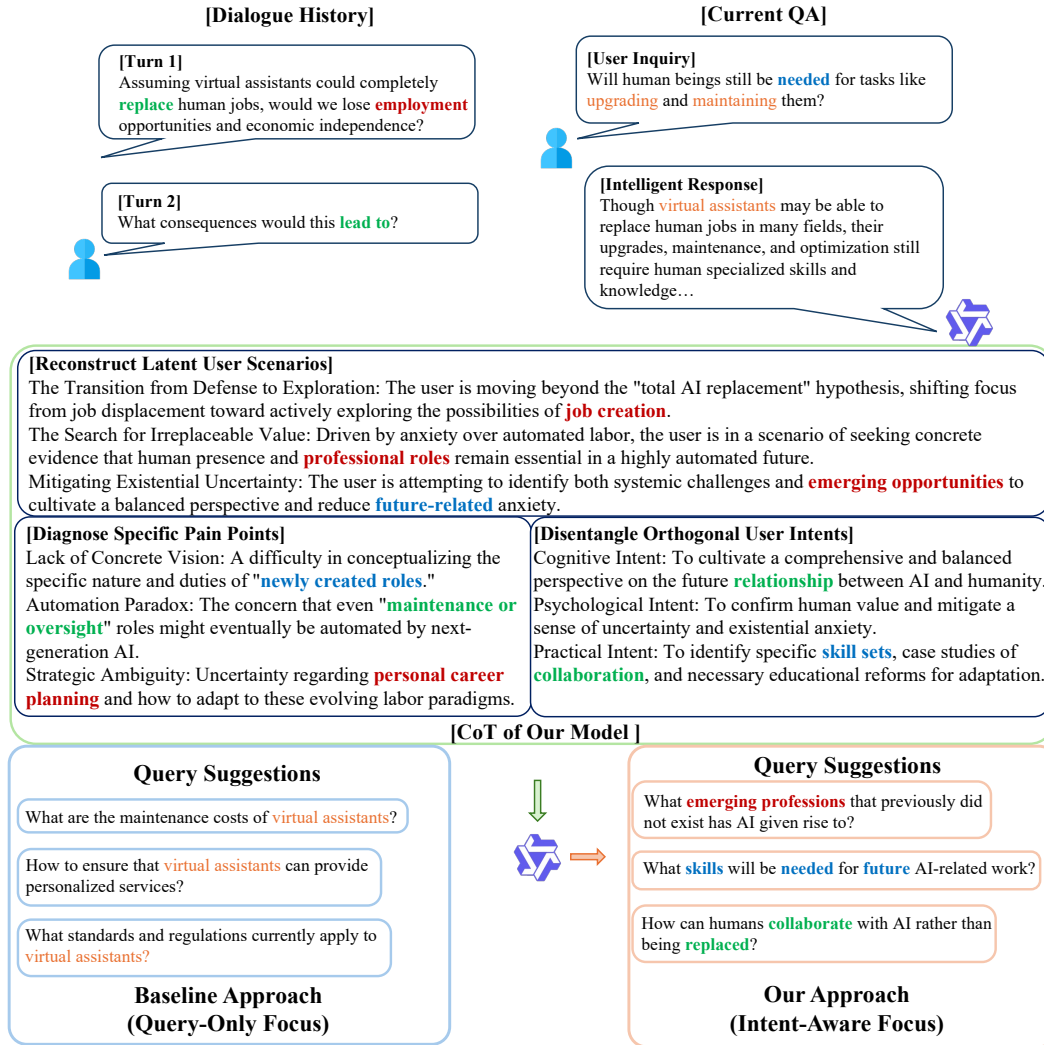


Figure 7: Case study comparing intent modeling approaches between baseline and our model.

Criterion	Description
Content Quality	
Timeliness Issue	The recommended query is outdated and no longer meets current requirements.
Security & Compliance	Content involving pornography, politics, violence, or illegal activities is deemed non-compliant.
Question Duplication	Recommended queries should avoid significant (over 80%) textual duplication.
History Duplicate	Overlapping less than 50% with historical queries, offering additional information gain.
Duplicate-of-Response	Overlapping less than 50% with previous responses and providing additional information.
Requirement Satisfaction	
Irrelevant	The query severely deviates from the user's current primary needs and intent.
Over-Assumption	Makes unfounded assumptions about the user's identity, context, or expertise.
Over-Reliance	Overly focuses on past conversations, neglecting current needs. Prioritize the current round.
Text Quality	
Word-Count-Exceeded	Exceeds 20 characters (Chinese, English x2, digits, punctuation, spaces each count as 1).
Unit Error	Non-standard units not commonly used, increasing user comprehension difficulty.
Poorly-Expression	The text is unclear, poorly expressed, with grammatical errors, typos, or ambiguities.
Inappropriate Topic	Pose questions about trending topics conversationally—avoid simply listing news.
Information Gain	
Logical Confusion	The query is logically inconsistent or contradictory with the preceding content.
Generic Question	The query is too vague, generic, and common knowledge, offering little added value.

Table 9: Human evaluation standards for query suggestion quality assessment.