Reinforcement Learning for Large Language Model Fine-Tuning: A Systematic Literature Review

Lingxiao Kong

lingxiao.kong@fit.fraunhofer.de

Fraunhofer Institute for Applied Information Technology FIT https://orcid.org/0009-0003-1968-7025

Qusai Ramdan

University of Southern Denmark, Denmark https://orcid.org/0000-0001-8159-918X

Oussama Zoubia

University of Cologne https://orcid.org/0000-0002-7930-7157

Jahid Hasan Polash

Fraunhofer Institute for Applied Information Technology FIT https://orcid.org/0009-0000-0099-9238

Mayra Elwes

University of Cologne https://orcid.org/0009-0005-9454-7174

Mehdi Akbari Gurabi

Fraunhofer Institute for Applied Information Technology FIT https://orcid.org/0000-0002-1734-8367

Lu Jin

University of Siegen

Ekaterina Kutafina

University of Cologne https://orcid.org/0000-0002-3430-5123

Roman Matzutt

Fraunhofer Institute for Applied Information Technology FIT https://orcid.org/0000-0002-4263-5317

Yuanbin Wang

University of Cologne https://orcid.org/0000-0003-1856-5205

Junqi Xu

Soochow University

Oya Deniz Beyan

Fraunhofer Institute for Applied Information Technology FIT https://orcid.org/0000-0001-7611-3501

Cong Yang

cong.yang@suda.edu.cn

Soochow University https://orcid.org/0000-0002-8314-0935

Zeyd Boukhers

zeyd.boukhers@fit.fraunhofer.de

Fraunhofer Institute for Applied Information Technology FIT https://orcid.org/0000-0001-9778-9164

Systematic Review

Keywords: Reinforcement Learning, Large Language Model, Fine-Tuning Techniques, Training Framework, Reward Modeling

Posted Date: November 27th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-8196796/v1

Additional Declarations: The authors declare no competing interests.

Reinforcement Learning for Large Language Model Fine-Tuning: A Systematic Literature Review

LINGXIAO KONG, Fraunhofer Institute for Applied Information Technology FIT, Germany
QUSAI RAMADAN, University of Southern Denmark, Denmark
OUSSAMA ZOUBIA, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany
JAHID HASAN POLASH, Fraunhofer Institute for Applied Information Technology FIT, Germany
MAYRA ELWES, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany
MEHDI AKBARI GURABI, Fraunhofer Institute for Applied Information Technology FIT, Germany
LU JIN, University of Siegen, Germany
EKATERINA KUTAFINA, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany
ROMAN MATZUTT, Fraunhofer Institute for Applied Information Technology FIT, Germany
YUANBIN WANG, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany
JUNQI XU, Soochow University, China
OYA DENIZ BEYAN, Fraunhofer Institute for Applied Information Technology FIT, Germany
CONG YANG, Soochow University, China
ZEYD BOUKHERS, Fraunhofer Institute for Applied Information Technology FIT, Germany

Large Language Models (LLMs) have been developed for a wide range of language-based tasks, while Reinforcement Learning (RL) has been primarily applied to decision-making problems such as robotics, game theory, and control systems. Nowadays, these two paradigms are integrated through different synergies. In this literature review, we focus on *RL4LLM fine-tuning*, where RL techniques are systematically leveraged to fine-tune LLMs and align them with various preferences. Our review provides a comprehensive analysis of 230 recent publications, presenting a methodological taxonomy that organizes current research into three primary method domains: *Optimization Algorithm*, concerning innovation in core RL update rules; *Training Framework*, regarding innovation in the orchestration of the training process; and *Reward Modeling*, addressing how LLMs learn and represent preferences and feedback. Within these primary domains, we further analyze methods and innovations through more granular categories to provide an in-depth summary of RL4LLM fine-tuning research. We address three research questions: 1) recent methods overview, 2) methodological innovations, and 3) limitations and future directions. Our analysis comprehensively demonstrates the breadth and impact of recent RL4LLM fine-tuning research while highlighting valuable directions for future investigation.

Additional Key Words and Phrases: Reinforcement Learning, Large Language Model, Fine-Tuning Techniques, Training Framework, Reward Modeling

Authors' Contact Information: Lingxiao Kong, Fraunhofer Institute for Applied Information Technology FIT, Germany, lingxiao.kong@fit.fraunhofer.de; Qusai Ramadan, University of Southern Denmark, Denmark, qura@mmmi.sdu.dk; Oussama Zoubia, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany, oussama.zoubia@uk-koeln.de; Jahid Hasan Polash, Fraunhofer Institute for Applied Information Technology FIT, Germany, jahidhasanpolash@gmail.com; Mayra Elwes, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany, mayra.elwes@uk-koeln.de; Mehdi Akbari Gurabi, Fraunhofer Institute for Applied Information Technology FIT, Germany, mehdi.akbari.gurabi@fit.fraunhofer.de; Lu Jin, University of Siegen, Germany, lu.jin@student.uni-siegen.de; Ekaterina Kutafina, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany, ekaterina.kutafina@uni-koeln.de; Roman Matzutt, Fraunhofer Institute for Applied Information Technology FIT, Germany, roman.matzutt@fit.fraunhofer.de; Yuanbin Wang, University of Cologne, Faculty of Medicine and University Hospital Cologne, Germany, yuanbin.wang@uk-koeln.de; Junqi Xu, Soochow University, China, 2262404058@stu.suda.edu.cn; Oya Deniz Beyan, Fraunhofer Institute for Applied Information Technology FIT, Germany, oya.deniz.beyan@fit.fraunhofer.de; Cong Yang, Soochow University, China, cong.yang@suda.edu.cn; Zeyd Boukhers, Fraunhofer Institute for Applied Information Technology FIT, Germany, zeyd.boukhers@fit.fraunhofer.de.

1 Introduction

The emergence of Large Language Models (LLMs)[1] has marked a transformative milestone in Artificial Intelligence (AI), fundamentally reshaping machine learning capabilities, especially in Natural Language Processing (NLP). This field demonstrated unprecedented conversational abilities and sparked widespread adoption of AI-powered language technologies across industries and research domains. The foundation of these LLMs relies on supervised and self-supervised pretraining on vast text corpora. However, these paradigms are insufficient for creating AI systems that reliably align with human values and preferences. They optimize for statistical patterns, often resulting in outputs that are factually incorrect, harmful, or misaligned with user intentions [6, 74].

To advance the performance of LLMs, researchers have investigated the integration of Reinforcement Learning (RL) techniques with LLMs, such as Reinforcement Learning with Human Feedback (RLHF) [129]. This integration leverages RL's learning paradigm that has already demonstrated remarkable success in sequential decision-making tasks. Rooted in behavioral psychology and optimal control theory, RL enables agents to learn optimal behaviors through trial-and-error interactions with their environment [162]. Unlike traditional supervised learning with static datasets, RL's capacity for dynamic interaction and iterative improvement through feedback loops offers a natural framework for incorporating human preferences, safety constraints, and task-specific objectives in LLMs' behavior. The synergy between LLMs' representational capacity and RL's goal-directed learning represents a fundamental advancement in creating AI systems that can understand, generate human-like text, and optimize behavior according to complex objectives beyond simple pattern matching. At their intersection, *RL4LLM fine-tuning* refers to research that applies RL techniques for fine-tuning LLM parameters to further improve performance on NLP tasks [135].

Although we are witnessing increasing adoption of RL4LLM fine-tuning approaches, a comprehensive understanding of their methodological details and RL4LLM synergies is still lacking. In [180], the authors primarily examine mature, proprietary LLM systems like InstructGPT, GPT-4, Claude 3, and DeepSeek-R1, cataloging their architectural choices and training methodologies. While comprehensive in documenting state-of-the-art models, it focuses on established systems rather than dissecting underlying methodological innovations. Another work [220] narrows its scope to RL applications for enhancing reasoning capabilities in complex logical tasks, particularly mathematics and coding. It emphasizes reasoning-specific RL techniques, training data requirements, and the path toward artificial superintelligence through scaled reasoning. In [99], the authors take a lifecycle-oriented approach, comprehensively covering RL applications throughout the entire LLM development pipeline from pre-training through post-training and inference stages. It provides temporal breadth by documenting RL's role across different developmental phases.

In contrast, our work targets the academic research community by systematically investigating methodological innovations within RL4LLM fine-tuning. Rather than cataloging proprietary models, focusing on specific applications, or surveying entire development lifecycles, we organize novel approaches by method domains and innovations to understand how and why different techniques work, enabling researchers to comprehend how specific technical choices within fine-tuning methodologies drive improvements in model capabilities. We synthesize findings from 230 papers (2022 to September 2025) to provide an in-depth understanding of current approaches, methodological innovations, limitations, and future work in RL4LLM fine-tuning research, serving as a foundational reference that shapes both academic research and industrial implementation strategies.

As the main focus for analyzing the methodological innovations of recent RL4LLM fine-tuning research, we categorize this research into three method domains: *Optimization Algorithm* as the core RL update rule, *Training Framework* for the

[1]For consistency with contemporary literature, we use "LLM" throughout this paper to refer to language models of all scales.

orchestration of the training process, and *Reward Modeling* for modeling how to learn and represent preferences and feedback in LLMs. We analyze the methods and address their nuanced details through the following research questions:

- **RQ1. How do the three method domains in RL4LLM fine-tuning differ in their application focus, addressed challenges, and method properties?** We identify and categorize recent RL4LLM fine-tuning works into method domains, examining application tasks and target challenges across these categories. The method properties are quantified to understand the reproducibility, computational complexity, and human effort.
- **RQ2. What are the core methodological innovations within each method domain, and what patterns emerge in their technical contributions and practical applicability?** We systematically categorize methods by their innovations and analyze their methodological details, deriving patterns of distinct RL4LLM fine-tuning methods and examining how synergies between RL optimization and LLM capabilities contribute to overall effectiveness.
- **RQ3. What are the primary limitations across RL4LLM fine-tuning methods and what future work do researchers propose to address these limitations?** Through systematic analysis of limitations and future directions across the literature, we identify unresolved challenges, methodological gaps, and underexplored areas within each domain. We map these limitations and their proposed solutions with methodological innovations, providing a roadmap for future RL4LLM fine-tuning research.

We first elaborate on the RL4LLM fine-tuning concept in Section 2, followed by outlining our search and data extraction strategies for this systematic literature review. For the analysis results, Section 3.1 addresses RQ1 through an overview analysis of method domains, challenges, applications, and properties. We then analyze innovations in each method domain regarding RQ2 in Section 3.2, examining methodological details and patterns. Section 3.3 discusses RQ3 by summarizing limitations and future directions identified in the review. Finally, Section 4 synthesizes these findings and presents our perspective on RL4LLM fine-tuning development.

2 Research Methodology

In this section, we first introduce the foundational knowledge of RL4LLM fine-tuning grounded in the RL/LLM taxonomy [135]. We then describe our rigorous search strategy and structured data extraction process, conducted as a systematic literature review following PRISMA guidelines [118].

2.1 RL4LLM Fine-Tuning

In the RL/LLM taxonomy [135], authors introduce it by systematically categorizing the intersection of these two components into three distinct classes: RL4LLM, LLM4RL, and RL+LLM, as illustrated in Figure 1. *RL4LLM* encompasses studies where RL techniques are strategically leveraged to enhance the performance of LLMs on tasks fundamentally related to NLP. This approach treats the LLM as the primary agent that benefits from RL-based optimization. In *LLM4RL*, the relationship is inverted: an LLM serves as an auxiliary component that assists in training an RL model designed to perform tasks that are not inherently related to NLP, such as robotics control or game playing. Finally, in *RL+LLM*, both components operate within a unified planning framework as collaborative agents, where an LLM and an RL agent work in tandem without either component directly contributing to the training or fine-tuning of the other, maintaining their distinct roles while benefiting from their combined capabilities.

Building on the RL4LLM taxonomy, where the authors distinguish between fine-tuning (directly modifying LLM parameters) and prompt optimization (iteratively refining prompts without parameter changes), this work focuses specifically on the *RL4LLM fine-tuning* subcategory. While the original taxonomy subdivides fine-tuning based on

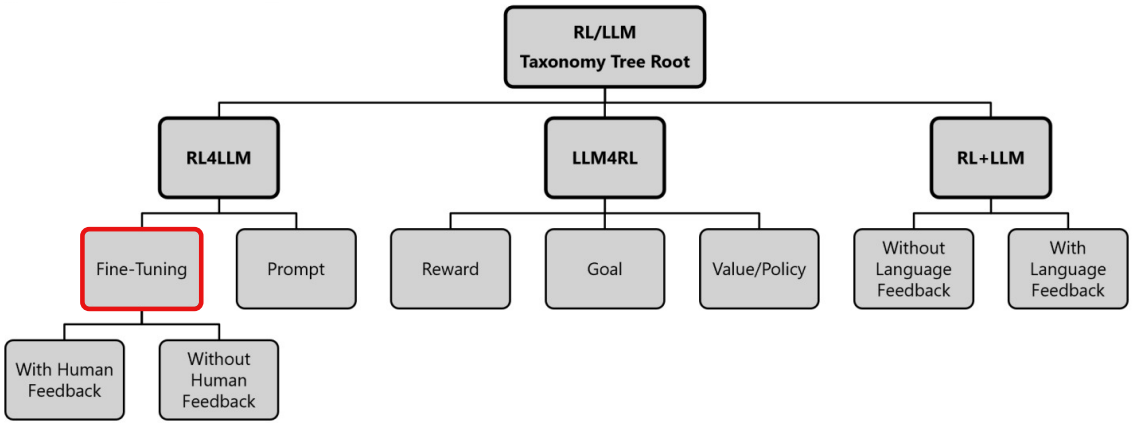

Fig. 1. RL/LLM taxonomy tree [135]. Our literature review focuses on RL4LLM fine-tuning.

feedback source, we adopt an alternative categorization approach and further refine the fine-tuning category by organizing methods according to specific method domains to better understand methodological innovations.

2.2 Search Strategy

Our search strategy includes four steps to refine the scope, as shown in Figure 2: identification from literature databases, preliminary screening, detailed eligibility assessment, and inclusion for data extraction.

Identification. We utilized Scopus[2] as our primary literature database due to its extensive, curated collection of high-quality, peer-reviewed scholarly content. To identify relevant publications, we constructed a search string requiring the terms *"LLM" OR "language model"* to appear in conjunction with *"RL" OR "reinforcement learning"* within the title, abstract, author keywords, or indexed keywords. We posited that papers developing RL4LLM methods at the intersection of these fields would explicitly mention these core concepts. Our use of the term "LLM" warrants clarification. While language models indicate a broad category encompassing various architectures like n-grams, RNNs, and LSTMs, LLMs are specifically modern variants defined by large scale and emergent abilities. However, contemporary literature frequently applies "LLM" to models of all sizes. For consistency, we use "LLM" throughout this paper to refer to language models of all scales. Additionally, we deliberately excluded broader related terms such as "NLP", "transformer", "reward model", and "reward function" from our search string, as these terms are frequently used in contexts unrelated to RL-LLM synergies and would unnecessarily broaden our literature scope. The search string is presented below:

```
( TITLE-ABS-KEY ( "LLM" ) OR TITLE-ABS-KEY ( "language model" ) )
AND ( TITLE-ABS-KEY ( "RL" ) OR TITLE-ABS-KEY ( "reinforcement learning" ) )
AND PUBYEAR > 2021 AND PUBYEAR < 2026 AND ( LIMIT-TO ( SUBJAREA , "COMP" ) )
AND ( LIMIT-TO ( DOCTYPE , "cp" ) OR LIMIT-TO ( DOCTYPE , "ar" ) ) AND
( LIMIT-TO ( SRCTYPE , "j" ) OR LIMIT-TO ( SRCTYPE , "p" ) ) AND ( LIMIT-TO ( LANGUAGE , "English" ) )
```

Besides the core search terms, we applied several filters to refine the results: the temporal scope was set from 2022 through September 2025 (the search was conducted on *September 24, 2025*); the subject area was limited to Computer

[2]https://www.scopus.com

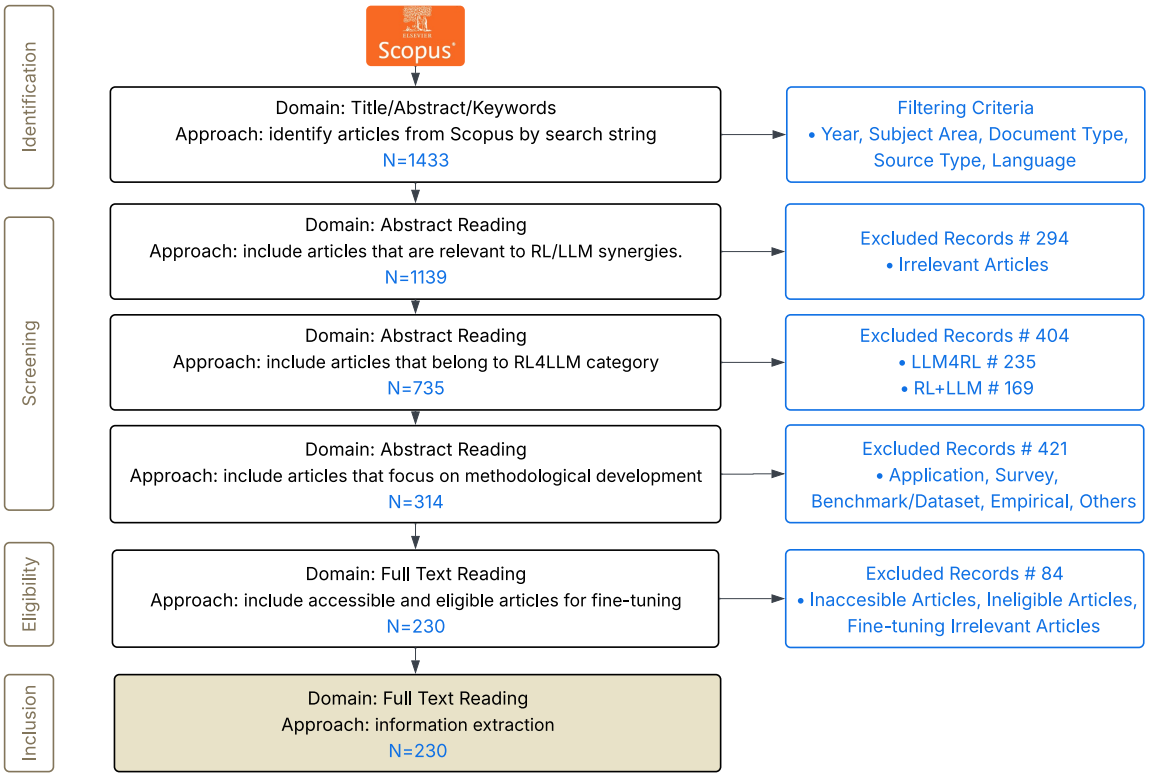

Fig. 2. Flow chart of the article search process with four stages adapted from PRISMA guidelines [118].

Science; the language was restricted to English; and the document types were confined to conference papers and articles from conference proceedings and journals. These constraints ensure domain relevance and focus on substantive, peer-reviewed research at the intersection of RL and LLMs. This initial search yielded *1433* papers.

Screening. The screening was conducted through systematic analysis of abstracts to efficiently evaluate each work's core contributions. This process consisted of three sequential stages: (1) an initial relevance screening, where we excluded 294 papers that mentioned RL and LLMs only as background context rather than focusing on their synergistic development; (2) a taxonomy-based classification using the RL/LLM taxonomy [135], which categorized remaining papers into RL4LLM (735), LLM4RL (235), and RL+LLM (169) domains, after which papers from the latter two categories were filtered out; (3) a focus on methodological developments, where we filtered out 421 papers concerning applications, surveys, benchmarks, or empirical analyses to concentrate exclusively on research advancing RL techniques within RL4LLM. Upon completing this screening, a final set of *314* papers was retained for in-depth eligibility assessment.

Eligibility. In the eligibility phase, we acquire all accessible papers. Then we conduct full-text reading by co-authors and LLMs, identifying fine-tuning and non-fine-tuning methods, which include inference-time techniques, prompt engineering, fine-tuning of auxiliary models, etc. Additionally, the eligibility assessment also filtered out underexplored works for irrelevant, non-RL4LLM, and non-methodological content, further covering gaps missed in the screening

phase. To ensure soundness for answering our three research questions, we also conducted an eligibility assessment following systematic review guidelines [73]. Each paper was evaluated against three predefined criteria:

- Does the paper describe its RL4LLM methodology, including associated challenges, applications, and properties?
- Does the method deliver impactful results that address the stated challenges and applications?
- Does the paper discuss or imply methodological limitations and future research directions?

The full-text review led to the exclusion of 84 papers that failed to pass the assessments, yielding a final corpus of *230* papers included for data extraction and synthesis of findings.

2.3 Data Extraction Strategy

Ten co-authors were involved in the data extraction process, with the 230 papers distributed evenly for each to review 23 papers without overlap. Co-authors brought diverse expertise from healthcare, data security, software engineering, computer vision, and machine learning, ensuring a multi-disciplinary perspective. Data was extracted from full text using a structured protocol (Table 1). To address potential inconsistencies from this multi-reviewer setup, we employed a *hybrid validation approach*: 1) LLM-assisted extraction to generate automated answers, 2) manual extraction by co-authors, and 3) conflict resolution by the main author, who reviewed both versions to resolve discrepancies. This method systematically enhanced the precision and consistency of review outcomes while reducing manual effort.

Table 1. Data extraction protocol for RL4LLM fine-tuning literature.

Research Questions	Topics	Question Type	Questions
RQ1. Recent Methods Overview	A.1 Key Contributions	Descriptive	What are the primary findings and main contributions of this work?
	A.2 Application Tasks	Descriptive	To which specific tasks or application domains is the method applied?
	A.3 Addressed Challenges	Descriptive	What specific challenges in RL4LLM does this work aim to solve?
	A.4 Reproducibility	Multi-Choice	To what extent are the proposed methods and experimental results reproducible? *(high/medium/low)*
	A.5 Computational Complexity	Multi-Choice	What is the computational resource usage of the proposed methods? *(high/medium/low)*
	A.6 Human Effort	Multi-Choice	What amount of human effort is required to implement the proposed methods? *(high/medium/low)*
RQ2. Methodological Innovations	B.1 Method Summary	Descriptive	Summarize the core methodology proposed in the paper and highlight the innovations.
	B.2 RL Formulation	Descriptive	Which RL algorithms are used, and how is the MDP (state, action, reward) formulated?
	B.3 LLM Foundation	Descriptive	Which base LLM(s) are used, and what are their key characteristics (e.g., size, type)?
	B.4 Integration Process	Descriptive	What is the detailed process for integrating the RL and LLM components?
	B.5 Synergies	Descriptive	What synergies or advantages does the RL4LLM integration provide?
	B.6 Experimental Results	Descriptive	What are the empirical results of the proposed method, and how effectively does it address the stated challenges?
RQ3. Limitations and Future Directions	C.1 Limitations	Descriptive	What limitations of the proposed method do the authors acknowledge?
	C.2 Future Directions	Descriptive	What future research directions does the paper suggest?

For the LLM-assisted extraction, we employed the Claude Sonnet 4[3] model as the extraction tool. This hybrid approach aligns with emerging trends in academic research methodology, as demonstrated by [145] in their investigation of LLM-assisted review processes. Their research revealed that numerous contemporary research projects increasingly utilize LLMs for automated review workflows. By leveraging LLMs, we efficiently captured the core content of each article while combining the results with human-curated review documentation to ensure accuracy and consistency. The final data extraction results are available in the GitHub repository[4].

3 Results

In this section, we analyze the extracted results from the data extraction process and organize our analysis around three key aspects of the 230 RL4LLM fine-tuning papers: (RQ1) recent methods overview, (RQ2) methodological innovations, and (RQ3) limitations and future directions. Section 3.1 addresses RQ1 by presenting an overview of the distribution of papers across three primary method domains: Optimization Algorithm, Training Framework, and Reward Modeling. We characterize papers in each domain according to their application tasks, addressed challenges, and key method properties, including reproducibility, computational complexity, and human effort requirements. Section 3.2 addresses RQ2 by providing detailed analyses for each method domain. We break down the methods into three-level hierarchies to provide a clear understanding and analyze the patterns of the RL4LLM fine-tuning methodological innovations. In Section 3.3, we focus on RQ3 and use the limitations and future directions recorded in data extraction to synthesize promising research directions, highlighting valuable avenues for advancement in each method domain.

3.1 Recent Methods Overview (RQ1)

How do the three method domains in RL4LLM fine-tuning differ in their application focus, addressed challenges, and method properties? We analyze the first block of extracted information (RQ1) *A.1-A.6* as shown in Table 1, including key contributions, application tasks, addressed challenges, as well as reproducibility, computational complexity, and human effort, to provide an overview of recent methods.

Method Domains. Based on the *A.1 Key Contributions*, we categorize the 230 papers into three method domains: *Optimization Algorithm (80 papers)*, *Training Framework (61 papers)*, and *Reward Modeling (89 papers)*. The close distribution across these domains indicates a balanced research focus in the RL4LLM fine-tuning method development.

Application Tasks. We analyze the extracted information from *A.2 Application Tasks* to examine the distribution of NLP applications across the three method domains. Since many papers address multiple tasks, we count each paper's contribution to every task it addresses as $\frac{\text{task occurences}}{\text{domain total}}$, allowing papers to be counted multiple times across different task categories. The 11 application tasks include: *Text Classification* (sentiment analysis, topic categorization), *Structured Information Extraction* (relation and event extraction, sequence labeling), *Reasoning and Inference* (natural language inference, commonsense reasoning), *Question Answering* (extractive and generative QA), *Summarization* (abstractive and extractive), *Dialogue Systems* (chatbots, conversational AI), *Open-ended Generation* (creative writing, story generation, text continuation), *Machine Translation*, *Text Paraphrasing* (style transfer, controlled text rewriting), *Code Generation* (program synthesis from natural language), and *Multimodal Tasks* (vision-language, speech-text integration).

Regarding the distribution of application tasks across method domains, we observe distinct patterns in Figure 3 that reflect the application preferences of different methods. Natural language understanding tasks are predominantly addressed by Training Frameworks, with 8.20% of works involving *Text Classification* and 34.43% involving *Reasoning*

[3]https://www.anthropic.com/claude/sonnet
[4]https://github.com/engineerkong/SLR-RL4LLM-Fine-Tuning-Methods.git

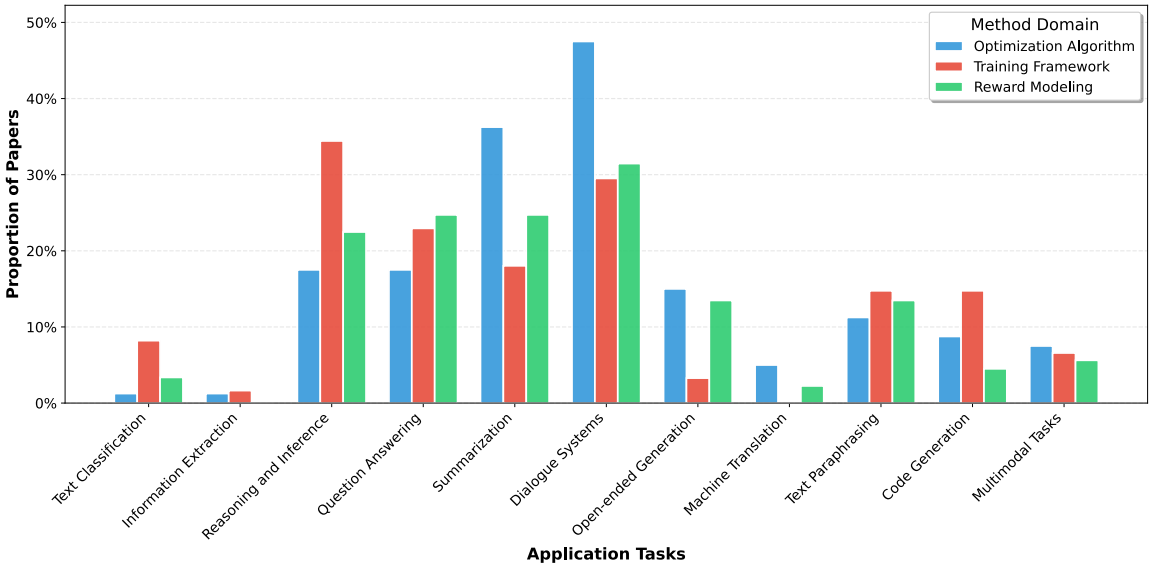

Fig. 3. Distribution of application tasks across RL4LLM fine-tuning method domains.

and Inference, followed by Reward Modeling and Optimization Algorithm. Notably, *Information Extraction* receives minimal attention across all domains. Natural language generation tasks have much higher occurrences overall. *Question Answering* appears in 24.72% of Reward Modeling works, followed by Training Framework (22.95%) and Optimization Algorithm (17.50%). In *Summarization* and *Dialogue Systems*, Optimization Algorithms show impressive presence, appearing in 36.25% of their works for Summarization and 47.50% for Dialogue Systems, significantly exceeding Reward Modeling and Training Framework. *Open-Ended Generation* receives modest coverage across all domains, with Training Framework showing the least presence. *Machine Translation* is limited, with only Optimization Algorithm and Reward Modeling showing applications. *Text Paraphrasing* receives moderate coverage across method domains, ranging from 11.25% to 14.75%. *Code Generation* appears most frequently in Training Framework works, while *Multimodal Tasks* receive low coverage across all domains. These analyses indicate that different application tasks are targeted across different method domains, with each domain exhibiting distinct patterns in its task applications.

Addressed Challenges. The scope of addressed challenges is more diverse than application tasks, with papers typically addressing multiple challenges simultaneously (often 3 or more). Based on extracted information from *A.3 Addressed Challenges*, we categorize the primary challenges into 12 domains: *Output Quality* focuses on improving generation quality and task-specific performance, including inconsistency and alignment tax. *Computational Efficiency* encompasses training time, convergence speed, and resource utilization. *Data Efficiency* addresses various data requirements (where "data" in RL4LLM primarily refers to reward and human or AI feedback), sample efficiency, annotation costs, and credit assignment. *Training Stability* covers hyperparameter sensitivity, mode collapse, convergence issues, catastrophic forgetting, and exploration-exploitation trade-offs. *Scalability and Generalization* examines the ability to scale across models and generalize to different tasks and domains, including distribution shift. *Safety and Bias* includes toxicity,

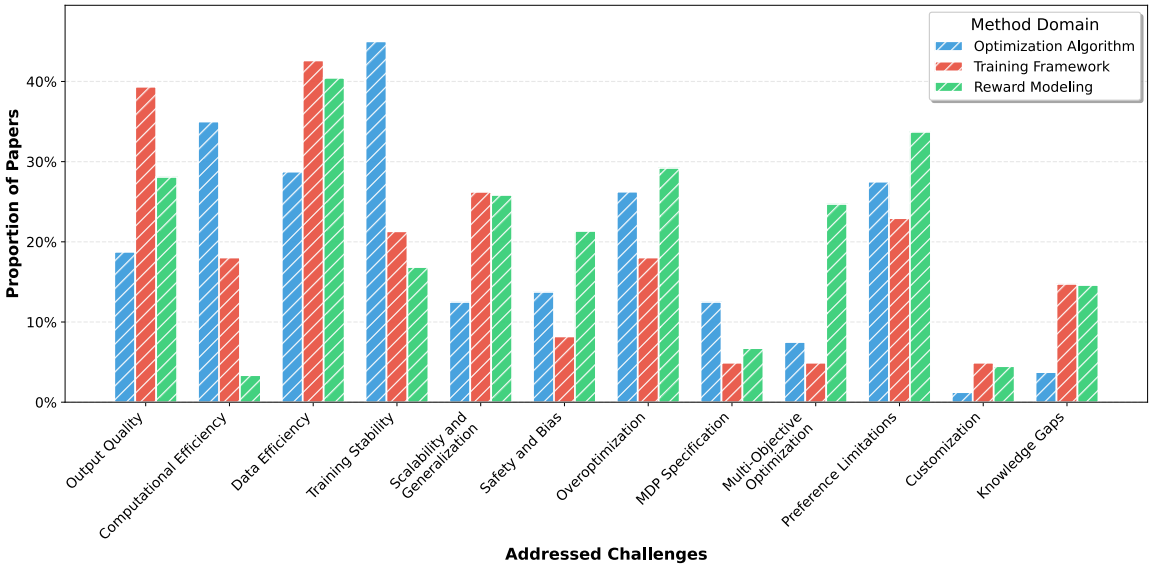

Fig. 4. Distribution of addressed challenges across RL4LLM fine-tuning method domains.

privacy, fairness, and harmful content prevention. *Overoptimization* addresses reward hacking, overfitting, and KL-regularization strategies. *MDP Specification* involves challenges in formulating state, action, and reward spaces. *Multi-Objective Optimization* focuses on balancing conflicting objectives and rewards such as helpfulness and safety. *Preference Limitations* encompasses diverse and inconsistent human preferences, noisy feedback, misalignment between learned and true preferences, and drawbacks of preference models. *Customization* addresses user-specific adaptation. *Knowledge Gaps* covers factual limitations, hallucinations, external knowledge integration requirements, and interpretability challenges. Similarly, we analyze the proportion of papers addressing each challenge as $\frac{\text{challenge occurrences}}{\text{domain total}}$.

As demonstrated in Figure 4, the distribution of addressed challenges reveals distinct patterns across the three method domains. *Output Quality* is addressed extensively by Training Framework (39.34% of works) but minimally by Optimization Algorithm (18.74%). Conversely, *Computational Efficiency* is heavily prioritized by Optimization Algorithm (35% of works), creating a substantial gap compared to Training Framework (18.04%) and Reward Modeling (only 3.37%). *Data Efficiency* receives considerable attention across all domains, with Training Framework leading at 42.62%. *Training Stability* is predominantly addressed by Optimization Algorithm, with an outstanding 45% of works focusing on this challenge, while the other two domains address it in only around 20% of their works. For *Scalability and Generalization*, Training Framework and Reward Modeling show similar attention levels, while Optimization Algorithm addresses these challenges at roughly half the rate. *Safety and Bias* concerns are uniquely emphasized by Reward Modeling (21.35% of works), with minimal attention from other domains. *Overoptimization* is most frequently addressed by Reward Modeling and Optimization Algorithm, with relatively lower focus in Training Framework. *MDP Specification* receives limited attention overall, with Optimization Algorithm showing the highest proportion at 12.5%. *Multi-Objective Optimization* is not widely addressed except for an impressive 24.72% in Reward Modeling. *Preference Limitations* are broadly addressed across domains, with Reward Modeling leading at 33.71%. *Customization* remains relatively unaddressed across all domains, while *Knowledge Gaps* show low attention from Optimization Algorithm but higher focus from the other two

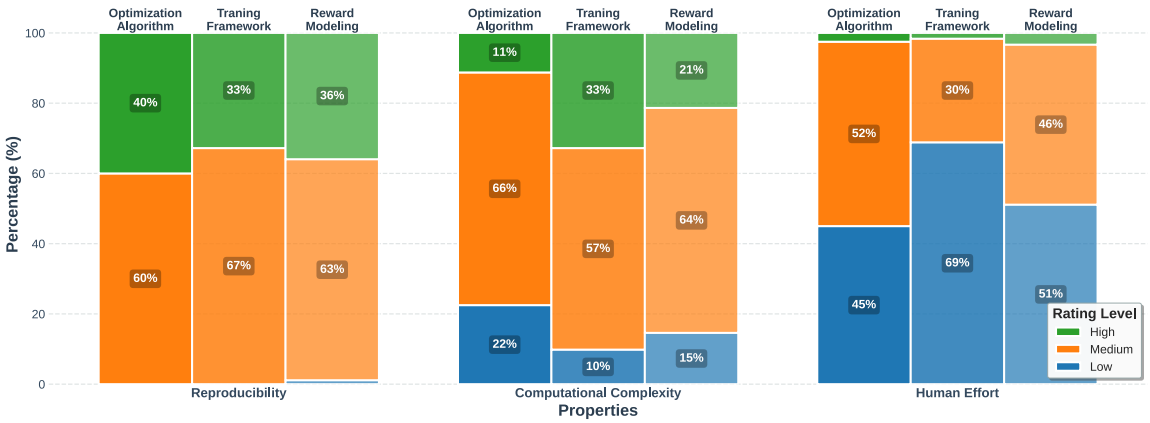

Fig. 5. Analysis of the RL4LLM fine-tuning method properties.

domains (approximately 15% each). These results explicitly illustrate the varying emphasis different method domains place on addressing current challenges, highlighting their complementary strengths and focus areas.

Method properties Finally, we analyze the extracted properties of RL4LLM fine-tuning methods according to the extracted information *A.4 Reproducibility*, *A.5 Computational Complexity* and *A.6 Human Effort*, as demonstrated in Figure 5. These properties are measured on a three-level scale: High, Medium, and Low.

Reproducibility depends on two factors: (1) resource availability, including code, models, and datasets, as well as open accessibility to evaluation models (proprietary models like GPT require paid access); and (2) implementation details, including descriptions of experimental settings, random seeds, and hyperparameter tuning procedures. This metric reflects how open and verifiable the reported results of RL4LLM fine-tuning methods are. Note that computational resources and human evaluation costs required for reproduction are not considered in this assessment. The distribution patterns across method domains are nearly identical, with 33-40% of papers achieving high reproducibility (satisfying both factors), while the remaining 60-67% achieve medium reproducibility (satisfying only one factor). Only very few papers exhibit low reproducibility across all domains. Notably, according to the extracted information, approximately half of the papers do not provide open access to their code, models, or datasets, or rely on proprietary resources, highlighting the need for greater transparency and openness in RL4LLM research publication practices.

Since RL fine-tuning is inherently computationally intensive, we assess *Computational Complexity* based on computational resource usage and the complexity of RL4LLM procedures, models, and dataset sizes. We categorize the complexity as follows: High complexity involves extremely long training times (many days), intensive resource requirements, and heavy or multi-stage fine-tuning procedures; Medium complexity involves moderate training times (hours to days), moderate resource usage, and standard fine-tuning tasks; Low complexity involves minimal training time, low computational resource requirements, and lightweight fine-tuning procedures. According to the extracted information, Training Framework methods exhibit the highest computational complexity, with 33% rated as high and only 10% as low, likely due to their iterative and hierarchical training procedures. Reward Modeling follows with the second highest computational complexity, reflecting the substantial computational effort required for modeling preferences and training reward models. Optimization Algorithm methods demonstrate the lowest computational complexity, with

only 11% high, 66% medium, and 22% low, likely because many algorithmic innovations specifically aim to reduce computational costs and require less extensive computational overhead.

We also assess *Human Effort*, which depends on the extent of human involvement in annotations, evaluations, domain expertise requirements, and manual setup and configuration for development. We categorize human effort as follows: High involves extensive human participation with many annotators curating large-scale datasets and conducting comprehensive evaluations; Medium involves moderate human involvement, such as a few people working to annotate smaller datasets, evaluate results, or conduct some domain investigation and setup; Low involves minimal or no additional human effort beyond the authors' research, development, and experimentation work. The RL4LLM fine-tuning papers we collected generally require medium-to-low human effort, with only a few papers requiring high effort. Optimization Algorithm methods exhibit the highest human effort requirements, with 52% of papers at medium level, likely due to extensive manual setup and configuration for algorithmic development. Training Framework methods demonstrate the lowest human effort requirements, with only 30% medium and 69% low, showing a clear inverse pattern with their computational complexity. Reward Modeling methods show moderate human effort with 46% at medium level, reflecting the substantial human annotation and evaluation effort needed for preference alignment.

RQ1: *How do the three method domains in RL4LLM fine-tuning differ in their application focus, addressed challenges, and method properties?*

Summary for RQ1: *These three method domains exhibit distinct patterns across all analyzed dimensions. In terms of application focus, Optimization Algorithms primarily target summarization and dialogue systems, Training Frameworks emphasize understanding tasks such as reasoning and text classification, while Reward Modeling demonstrates comprehensive distribution across various tasks. Regarding addressed challenges, Optimization Algorithms emphasize computational efficiency and training stability, Training Frameworks focus on output quality and data efficiency, while Reward Modeling methods prioritize reward-oriented challenges such as multi-objective optimization and preference limitations. For method properties, all domains achieve moderate-to-high reproducibility, though improvements in research openness remain needed. Optimization Algorithms demonstrate the lowest computational complexity but the highest human effort requirements, Training Frameworks exhibit the highest computational complexity but the lowest human effort, while Reward Modeling shows moderate levels for both properties. These patterns clearly indicate specialized developmental aims for distinct approaches to RL4LLM fine-tuning.*

3.2 Methodological Innovations (RQ2)

RQ2. What are the core methodological innovations within each method domain, and what patterns emerge in their technical contributions and practical applicability? We analyze the extracted information *B.1-B.6* in Table 1 by carefully examining the information and categorizing the methods using a top-down approach to create a three-level methodology hierarchy as presented in Tables 2, 3, and 4.

3.2.1 Optimization Algorithm. These algorithm-oriented methods can be categorized according to their underlying RL algorithms into: 1) Policy Gradient and Actor-Critic, 2) Direct Policy Optimization, 3) Value-Based Learning, 4) Game Theory, and 5) Distributional Matching algorithms. We then analyze each category by identifying innovations.

Policy Gradient and Actor-Critic. These algorithms optimize LLMs by framing output as sequential decision-making, where the policy (LLM) selects tokens to maximize cumulative reward from human preference models or task-specific metrics. Policy Gradient methods directly optimize policy parameters using gradient ascent on expected rewards, with REINFORCE as the foundational algorithm. Actor-Critic architectures enhance this through dual components: the

Table 2. Review of Optimization Algorithm innovations for RL4LLM fine-tuning.

Optimization Algorithm	Innovation	Reference Papers
Policy Gradient and Actor-Critic	PPO-Based Actor-Critic	RLHF [129], MA-RLHF [14], BSPO [29], PyTester [163], CPPO [216], ReMax [93], DfPO [62], Safe RLHF [28], APA [79], MTPO [148], POAD [183], GPT-Critic [63], STR [208], PIT [181]
	Pure Policy Gradient	BindGPT [231], ESRL [171], REBEL [43], CodeRL [80], RAINER [98], NLPO [140], TrufLL [115], Elastic Reset [128]
	Alternative Actor-Critic Variants	TOLE [91], CRT [54], REFUEL [44]
	Hybrid RL Formulations	CoPG [41], ZPG [222], A-LOL [4], LIRE [240], f-DPG [45]
Direct Policy Optimization	Divergence Modifications	χPO [57], f-DPO [169], CAN [60], VPO [13], DIL [198]
	Enhanced Loss Functions	ChatTune-DPO[68], SamPO [107], MMPO [71], Length-regularized DPO [132], ODPO [2], Mallows-DPO [20], D^{2}PO [149], DPO [139], VCB [114], AMP [221], RS-DPO [70], fDPO [119], ADPO [65], RDO [179], QDPO [82]
	Multi-Response Extensions	MPPO [199], DMPO [152], MODPO [237]
	Weight Modifications	IW-DPO [105], WPO [235], GRPO [142]
	Non-DPO Methods	PRO [156], RRHF [214], BFPO [226], SELM [224], IPO-MD [10], C-RLFT [172]
Value-Based Learning	Direct Q-Learning Variants	Q-SFT [53], ILQL [155], SQL for Text Generation [47]
	Residual and Adaptive Q-Learning	Q-Adapter [92]
	Inverse Q-Learning	Inverse-Q* [196], IQLearn [194]
Game Theory	Online Mirror Descent	INPO [228], SPPO [192]
	Magnetic Mirror Descent	MPO [175]
	Nash Equilibrium	NLHF [123]
Distributional Matching	Best-of-N	vBoN [1], BOND [146], RSO [103], BoN-SFT [27]
	GFlowNet	GDPO [77], Armotimized GFlowNets [56]
	Importance Sampling	BRAIN [130], EXO [64]

actor (policy network) selecting actions and the critic (value function) estimating expected rewards, reducing gradient variance for stable learning. Proximal Policy Optimization (PPO) advances this paradigm by introducing clipping mechanisms that constrain policy updates, preventing destabilizing changes while maintaining sample efficiency. Evolution of these 30 methods for LLMs reveals significant innovations across four key algorithmic categories.

PPO-Based Actor-Critic innovations include Macro-Action RLHF (MA-RLHF) [14], which introduces temporal abstraction through token sequence grouping for enhanced credit assignment, and Behavior-Supported Policy Optimization (BSPO) [29], which pioneers value regularization via behavior-supported Bellman operators to address reward overoptimization. *Pure Policy Gradient* innovations employ REINFORCE variants for specialized applications, exemplified by BindGPT [231]'s molecular design framework, achieving 100× speedup through docking score optimization, and Efficient Sampling-based Reinforcement Learning (ESRL) [171], which enhances sample efficiency through advanced sampling strategies. *Alternative Actor-Critic Variants* explore novel architectures, particularly Curiosity-driven Red-Teaming (CRT) [54], which employs intrinsic motivation for systematic safety vulnerability discovery through exploration-based policies. *Hybrid RL Formulations* establish new theoretical foundations by combining RL with other paradigms. Contrastive Policy Gradient (CoPG) [41] bridges supervised learning stability with RL optimization through contrastive objectives, while Zeroth-order Policy Gradient (ZPG) [222] eliminates gradient computation via derivative-free optimization. These approaches represent emerging trends addressing fundamental challenges in stability, efficiency, and safety across diverse LLM training scenarios.

Direct Policy Optimization. This subclass represents a paradigm shift in RLHF by reducing explicit reward model training. Direct Policy Optimization methods directly optimize LLM policies using preference data, with Direct Preference Optimization (DPO) serving as the foundational algorithm. DPO leverages the Bradley-Terry model to reformulate the RL objective as a classification problem, employing a loss function based on the log-sigmoid of the scaled reward difference between preferred and rejected responses. This approach bypasses explicit reward model training while preserving the theoretical rigor of preference-based optimization. Some DPO variants also employ reward models, seeking to combine the stability and interpretability of explicit reward signals with DPO's efficiency. Our analysis categorizes the 32 works into five major innovation directions.

Divergence Modifications directly modify the fundamental divergence constraint, exemplified by χ^2-Preference Optimization (χPO) [57], which substitutes DPO's reverse KL divergence with chi-squared divergence for reduced overoptimization. *Enhanced Loss Functions* augment the standard objective with additional regularization, such as ChatTune-DPO [68], which incorporates contrastive penalties and entropy regularization to prevent mode collapse in low-resource scenarios. *Multi-Response Extensions* generalize pairwise frameworks to handle multiple responses, as seen in Multi Pair-wise Preference Optimization (MPPO) [199], which processes N+1 responses using the geometric mean of token likelihoods and achieves significant MT-Bench improvements. *Weight Modifications* address distribution shifts through adaptive reweighting, exemplified by IW-DPO [105], which estimates density ratios for test distribution optimization. *Non-DPO Methods* explore alternative frameworks beyond Bradley-Terry models, such as Preference Ranking Optimization (PRO) [156], which implements ranking-based preference learning with active query selection, achieving comparable alignment with 50% fewer human evaluations. These innovations advance Direct Policy Optimization from simple pairwise optimizers to sophisticated frameworks handling real-world deployment challenges.

Value-Based Learning. These algorithms focus on estimating the optimal value function, which represents the maximum expected future reward an agent can achieve from any given state. Q-Learning learns the action-value function (Q-function) to evaluate the quality of taking specific actions in particular states, with the agent's policy derived by selecting the action with the highest estimated value. Our analysis identifies 6 works in this domain. Innovation unfolds primarily through *Direct Q-Learning Variants*, where methods like Q-SFT [53] reformulate classical Q-learning as weighted cross-entropy losses, treating token probabilities as conservative value estimates and embedding Bellman updates directly into supervised fine-tuning objectives. *Residual and Adaptive Q-Learning* approaches like Q-Adapter [92] leverage residual Q-learning frameworks to learn incremental value function adjustments, establishing direct correspondence between Q-functions and reward differences to enable preference learning without explicit reward modeling. *Inverse Q-Learning* techniques, exemplified by IQLearn [194], reformulate inverse RL as maximum likelihood estimation with temporal difference regularization, extracting reward functions from expert demonstrations for subsequent value-based optimization. The literature reveals convergence towards hybrid approaches that embed classical Q-learning principles within supervised learning frameworks, combining the temporal compositionality of value-based RL with the training stability of supervised learning.

Game Theory. These algorithms model strategic interactions between rational decision-makers, with Nash Equilibrium representing stable states where no player benefits from unilateral strategy changes. In LLM alignment, these methods frame preference learning as strategic interactions between policies, seeking stable Nash solutions. Our literature review contains 4 papers on innovations in this specific RL algorithm. *Online Mirror Descent* (OMD) generalizes gradient descent using mirror maps and Bregman divergences for superior convergence on non-Euclidean geometries. Iterative Nash Policy Optimization (INPO) [228] applies OMD with entropy regularization to symmetric two-player games, achieving superior performance through direct optimization over preference datasets without expected win rate computation.

Magnetic Mirror Descent extends OMD by introducing a magnetic term that periodically updates the reference policy to guide convergence toward the original Nash equilibrium. Magnetic Preference Optimization (MPO) [175] adapts this to RLHF as a two-player constant-sum game, achieving linear convergence rates that outperform standard mirror descent. *Nash-Equilibrium* approaches, such as Nash Learning from Human Feedback (NLHF) [123], bypass scalar rewards by learning pairwise preference models and computing Nash equilibria directly from comparisons, improving expressivity and robustness compared to reward-based methods. These innovations reveal that current Game-Theoretic LLM alignment research is dominated by mirror descent algorithms.

Distributional Matching. These methods reframe the core objective of RL4LLM, shifting from expected reward maximization to explicit distributional alignment. Instead of pursuing scalar rewards, the goal is to train a policy π_θ to minimize divergence from a target distribution π^* that encodes desired behavioral traits. This target distribution is typically derived through principled statistical algorithms such as Best-of-N, Generative Flow Networks (GFlowNets), and Importance Sampling techniques. We identified 8 papers concerning innovations in Distributional Matching methods. The *Best-of-N* paradigm constructs an empirical target distribution by sampling multiple responses from a base model and selecting those with the highest rewards, then distilling this distribution into a single efficient policy. Variational Best-of-N (vBoN) [1] formalizes this by fine-tuning an LLM to minimize reverse KL divergence to the analytically derived BoN distribution, preserving high-reward characteristics while eliminating inference-time sampling costs. *GFlowNets* enable diverse generation by sampling sequences with probability proportional to given rewards. Methods like GFlowNet Direct Preference Optimization (GDPO) [77] adapt this for offline alignment, deriving token-wise rewards from pairwise preference data and integrating them into GFlowNet's detailed balance objective to enhance output diversity. *Importance Sampling* techniques address intractable distribution matching through Monte Carlo estimation. Bayesian Reward-conditioned Amortized Inference (BRAIN) [130] uses Bayesian inference to define a target reward-conditioned posterior and introduces a self-normalized baseline in its gradient estimator, significantly reducing variance and improving performance. These methods demonstrate movement toward principled distribution matching with emphasis on theoretical guarantees, diversity preservation, and computational efficiency.

3.2.2 Training Framework. Training Framework approaches have core innovations in fundamental architectural designs. According to different architectures, we can categorize the works into four categories: 1) Sequential Pipeline, 2) Iterative Refinement, 3) Hierarchical Architecture, and 4) Online Adaptive Framework. Subsequently, we provide a comprehensive analysis of each category, exploring more granular innovation types within each framework, identifying patterns in Training Framework methods, and revealing insights into RL4LLM fine-tuning architectural design principles.

Sequential Pipeline. These methods decompose the monolithic process of model training and inference into distinct, specialized stages. Our analysis examines 12 research papers on Sequential Pipeline frameworks, focusing on how RL is integrated into multi-stage training and inference pipelines.

Sequential Pipeline Frameworks can be categorized into three distinct types based on their architectural patterns and functional objectives. *Data-Centric Pipelines* systematically improve data quality and then model performance through sequential stages, exemplified by the Reasoning Distillation Framework [144], which generates synthetic explanations, performs knowledge distillation, and applies RL refinement for knowledge transfer to smaller models, and WizardMath [110], which evolves mathematical problems, trains reward models, and optimizes through PPO to enhance mathematical reasoning. *Modular Component Integration* develops independent components and systematically integrates them into cohesive systems, demonstrated by Oreo [90], which trains modular context reconstruction components via SFT, integrates them through contrastive learning, and refines the complete pipeline using RL to align

Table 3. Review of Training Framework innovations for RL4LLM fine-tuning.

Training Framework	Innovation	Reference Papers
Sequential Pipeline	Data-Centric Pipelines	Reasoning Distillation Framework [144], WizardMath [110], RewriteLM [153], Teams-RL [46]
	Modular Component Integration	Oreo [90], RLAF [126]
	Capability Development Pipelines	UCP [184], PCT [112], ALMoST [72], BAMBINO-LM [151], HTL [89], DPPPO [185]
Iterative Refinement	Data-Centric Refinement	ILR [210], DICE [18], RAFT [33]
	Multi-Agent Iterative Systems	Reflect-RL [234], AutoIF [32], SPAG [26], WizardArena [109]
	Internal Self-Refinement	SCoRe [76], RLC [131], RISE [137], HIR [225]
	External Feedback Refinement	LeReT [55], ReST-MCTS* [215], StepCoder [35], AutoPRM [24]
	Exploration-Based Refinement	COPO [5], PRS [209], R^3 [195], SpannerSampling [42], Quark [108]
	Process-Level Refinement	SVPO [19], ALT [104], ReFT [167]
Hierarchical Architecture	Parallel Frameworks	FedRLHF [38], CORY [111], FedBis [186]
	Multi-Level Frameworks	ArCHer [236], KEHRL [84], MOS [189], LDPP [51]
	Hybrid Integration	UPO [3], SPO [86], DEFT [239], HyPO [157], RLVF [158], STG [67]
	Modular Extension Frameworks	PPDPP [31], TWOSOME [164]
Online Adaptive Framework	Real-time Feedback	CORGI [7], DRESS [23], APL [122], Online Strategic Feedback [49], SOTOPIA-π [178]
	Dynamic Knowledge Integration	RID [39], IM-RAG [202], PRewrite [75], SECOND THOUGHTS [101]
	Continuous Learning Systems	Asynchronous RLHF [127], Group-Invariant Policy [230]

context processing with generator preferences. *Capability Development Pipelines* build foundational abilities, align them with preferences, and specialize for deployment scenarios. UCP [184] establishes multi-task coding foundations through joint fine-tuning, aligns via SimPO-based RL, and specializes through self-infilling inference mechanisms, while the seminal RLHF approach [129] builds instruction-following foundations via SFT, aligns with human preferences through reward modeling, and specializes via PPO optimization. These sequential approaches demonstrate the effectiveness of staged development in achieving complex RL4LLM objectives through systematic progression.

Iterative Refinement. These frameworks progressively improve through cyclic feedback loops rather than sequential training, leveraging RL to create self-improving systems that enhance performance through various feedback mechanisms. Many frameworks operate in online settings where models learn from their own generated data while optimizing across multiple dimensions simultaneously. We evaluate 23 methods and categorize them into six categories.

Data-Centric Refinement treats training data as refinable artifacts, with ILR [210] revolutionizing RLHF by redirecting comparison feedback from model optimization to dataset improvement through cross-labeling. *Multi-Agent Iterative Systems* create collaborative refinement through multiple components, with Reflect-RL [234] implementing two-player refinement where reflection models guide policy improvements, and AutoIF [32] enabling self-play through execution feedback verification. *Internal Self-Refinement* enables self-awareness through sequential processing, with SCoRe [76] implementing inference-level self-correction cycles. *External Feedback Refinement* leverages external systems for improvement, with LeReT [55] implementing multi-hop retrieval refinement and StepCoder [35] utilizing compiler feedback. *Exploration-Based Refinement* systematically explores uncertain prompt-response regions, as COPO [5] integrates count-based exploration with Upper Confidence Bound bonuses for comprehensive preference data coverage. *Process-Level Refinement* implements granular reasoning refinement, with SVPO [19] generating step-level preferences and training explicit value models for reasoning step evaluation. These frameworks represent a shift from static training paradigms to self-improving systems that evolve through structured feedback loops.

Hierarchical Architecture. These frameworks address scalability, efficiency, and performance challenges through sophisticated system design, fundamentally transforming how RL training is orchestrated by distributing computational load across parallel processes or combining distinct methodological approaches. With 15 papers in our collection, we analyze these frameworks by examining their innovations in four distinct strategies.

Parallel Frameworks leverage distributed computing to enable simultaneous training across multiple agents or clients, decomposing RL4LLM processes into concurrent sub-processes with coordinated communication. FedRLHF [38] distributes RLHF across federated clients that contribute specialized local knowledge while benefiting from global model improvements, while CORY [111] employs a dual-agent architecture where pioneer and observer agents train simultaneously with complementary access. *Multi-Level Frameworks* decompose complex problems into multiple abstraction levels, enabling tractable optimization by separating concerns across temporal or conceptual scales. ArCHer [236] decomposes RL into utterance-level strategic decisions and token-level implementation, each optimized with complementary algorithms, while KEHRL [84] separates knowledge enhancement into entity detection and triple selection levels optimized through coordinated processes. *Hybrid Integration* paradigms combine heterogeneous training paradigms within unified frameworks. UPO [3] integrates preference optimization methods (KTO, DPO) with offline RL for auxiliary objectives, while HyPO [157] combines offline preference learning with online policy sampling to harness both data efficiency and adaptability benefits. *Modular Extension Frameworks* employ RL components as external modules that guide LLM capabilities without modifying core language model training. PPDPP [31] employs a pre-trained RoBERTa model as an external policy planner that predicts dialogue strategies for agents. These frameworks demonstrate how architectural decomposition and parallel coordination effectively address scalability and complexity in RL4LLM fine-tuning.

Online Adaptive Framework. Moving beyond static architectures, Online Adaptive frameworks introduce real-time learning, transforming LLMs from static models into adaptive agents. Through RL, they conduct continuous in-situ model updates, enabling simultaneous optimization for objectives like accuracy and safety based on live feedback. We categorize 11 instantiations into three subcategories based on their adaptation mechanisms.

Real-time Feedback enables models to receive and act upon feedback within single interaction sessions, with CORGI [7] exemplifying online adaptation by learning to improve responses iteratively rather than requiring retraining. *Dynamic Knowledge Integration* incorporates real-time decision-making during generation, with adaptive choices about knowledge retrieval, prompt optimization, or response strategies. RID [39] demonstrates this through real-time knowledge integration by making retrieval decisions dynamically during generation, enabling adaptive responses based on contextual information needs through rationale-aware explanation generation, dynamic switching, and explanation distillation modules. *Continuous Learning Systems* represent the most advanced form, enabling parameter updates during deployment without interrupting service. Asynchronous RLHF [127] addresses computational bottlenecks by decoupling inference and training phases, enabling online learning where models continuously adapt based on real-time interactions without interrupting user service, though this remains challenging and underexplored. These methods collectively push toward Online Adaptive Frameworks that evolve with user interactions.

3.2.3 Reward Modeling. In RL4LLM, Reward Modeling is a specific reward-oriented approach to improve LLMs. The innovation techniques can be structured based on their distinct reward modeling and design methodologies into five primary categories: 1) Reward Model Architecture, 2) Reward Granularity and Density, 3) Reward Bias Calibration, 4) Synthetic Preferences Generation, and 5) Compositional Rewards methods. We subsequently conduct a detailed examination of each category, investigating more nuanced innovation variants within each approach.

Table 4. Review of Reward Modeling innovations for RL4LLM fine-tuning.

Reward Modeling	Innovation	Reference Papers
Reward Model Architecture	Parameter Sharing Architectures	P-ShareLoRA [100], WARM [141]
	Mixture-of-Experts Architectures	ArmoRM [174], DMoERM [138], MoRE [124], MaxMin-RLHF [15], PoE-based Reward Modeling [150]
	Multi-Head Architectures	Constractive Goal-Conditioned Learning [125], GRM [207], ODIN [21]
	Memory-Augmented Architectures	Proto-RM [219], RLKF [94]
	External Component Integration	Themis [88], GazeReward [106], InfoRM [116]
Reward Granularity and Density	Token-Level Dense Rewards	Seq2seq RM [233], RELC [12], RLMEC [25], TLCR [211], ABC [16], Token-Level Reward Modeling [200], FINE-GRAINED RLHF [193]
	Step-Level Process Rewards	GraphPRM [134], PAV [147], ER-PRM [217], MATH-SHEPHERD [177], GLoRe [50], SENSEI [102]
	Multi-Granular Rewards	FGAIF [66], Implicit Toxicity Framework [182]
	Dense Reward Enhancement	KRLS [212]
Reward Bias Calibration	Post-hoc Bias Correction	RC-Mean/RC-LWR [61], CAA [197], Reward Dropout [81]
	Confidence-based Calibration	PPO-M/PPO-C [83], CALS [58], RCfD [143], Fact-RLHF [160], IDS [238], ADVPO [227], SaySelf [201], CONQORD [165]
	Distributional Methods	BIRL [9], IDS-based RLHF [136], LSAM [168], DPL [154], ENTFA [11], R^3M [8], CVaR [17], DRO [52]
Synthetic Preferences Generation	Cross-Modal Synthesis	RoVRM [170], MACAROON [190]
	Self-Evolutionary Loops	SER [59], UGDA [159], ALOE [191]
	Task-Performance Feedback	RLPF [187], RaFe [113], ExpCTR [213]
	Contrastive Automatic Labeling	RLCD [205], RLCF [34], Contrastive Reward Modeling [22]
	Knowledge-Guided Synthesis	OCEAN [188], DogeRM [97], ML-IRL [87], MupPCQA [203]
	Hybrid Multi-Component Synthesis	SALMON [161], Self-motivated Learning [40], DeMem [69]
Compositional Rewards	Dynamic Weighting Optimization	Multi-style Rewards [30], Fast RL [85], DYNAOPT [117], DPA [173], Constrained RLHF [120]
	Hierarchical Decomposition	ALARM [78], Semi-structured Explanation Rewards [48]
	Structure Reward Engineering	VeriSeek [176], Constraint-aware KBQA [204], RBRs [121], PCRM [232]
	Domain-Specific Decomposition	LLM2ER-EQR [206], FLAME [96], RELAX-based MDS [133], BackMATH [223], SYRELM [36], RLLR [95], TRUSTWORTHY-ALIGNMENT [229], HuatuoGPT [218], Reinforce-Detoxify [37], Moral Intrinsic Rewards [166]

Reward Model Architecture. This domain refers to the structural design, component organization, and computational arrangement of reward models, encompassing neural network architecture, parameter organization schemes, multi-component designs, and structural modifications that enable effective preference learning and reward prediction. We analyze 15 papers with relevant architectural innovations.

Parameter Sharing Architectures like P-ShareLoRA [100] introduce parameter sharing schemes where each user's reward function parameters are structured as $\Theta_i = \Theta^{init} + \Delta\Theta_i$, with $\Delta\Theta_i = BW_i$. This architecture shares matrix B across users while maintaining personalized W_i matrices, creating a hierarchical parameter organization that balances shared knowledge and personalization. *Mixture-of-Experts Architectures* introduce dynamic routing mechanisms that allow specialized processing paths based on input characteristics. DMoERM [138] exemplifies this through a dual-layer MoE architecture: an outer sparse MoE layer routing inputs to task-specific experts using frozen pre-trained routers, and an inner dense MoE layer decomposing tasks into capability dimensions with trainable routers. *Multi-Head Architectures* address the trade-off between specialization and generalization through dual-component designs. GRM [207] implements both a reward head that minimizes standard reward loss for preference prediction and an LM head that minimizes

token prediction loss to maintain language understanding, with shared hidden states enabling knowledge transfer between objectives. *Memory-Augmented Architectures* introduce dynamic adaptation capabilities through prototypical learning mechanisms, as demonstrated by Proto-RM [219], which incorporates encoding layers, dynamic prototype banks, and episodic memory refinement mechanisms enabling rapid adaptation to new preference patterns. *External Component Integration* methods like Themis [88] extend reward architecture beyond self-contained models by integrating external computational components through structured reasoning chains, representing a paradigm shift toward hybrid architectures that combine internal learned representations with external knowledge sources and tool capabilities. These architectural innovations demonstrate the evolution from monolithic reward models toward modular, adaptive systems that can handle diverse preference patterns and complex tasks.

Reward Granularity and Density. This domain refers to the temporal and spatial resolution at which rewards are assigned during learning. Granularity describes the temporal resolution of reward assignment, including token-level, step-level, and sequence-level approaches. Density describes the frequency and richness of reward signals, encompassing dense, sparse, and process rewards. The synergy between granularity and density enables more effective credit assignment, reducing the temporal credit assignment problem in long-sequence generation tasks. We analyze 16 recent papers and categorize their innovations into four types.

Token-Level Dense Rewards apply dense reward signals at the token generation level, enabling fine-grained credit assignment for each vocabulary decision. TLCR [211] introduces a token-level continuous reward mechanism that assigns real-valued rewards to each generated token based on human preferences, employing a specialized reward model that learns to predict token-level quality scores for dense supervision throughout generation. *Step-Level Process Rewards* decompose complex reasoning tasks into logical steps with specialized reward models that evaluate reasoning process quality rather than just final outcomes. GraphPRM [134] introduces automated step-level annotation for graph reasoning through task-oriented trajectories that translate algorithm execution into natural language steps, using Monte Carlo Tree Search for diverse path generation and training Process Reward Models to evaluate step-wise correctness. *Multi-Granular Rewards* simultaneously apply rewards at multiple granularity levels, creating hierarchical reward structures that capture both local and global quality aspects. FGAIF [66] implements a three-tier approach: (1) AI-based feedback collection using ChatGPT for atomic fact extraction and LLaVA for image consistency verification, (2) training specialized reward models for object existence, attribute accuracy, and relational correctness, and (3) RL optimization using combined multi-granular rewards. *Dense Reward Enhancement* enhances traditional reward modeling through auxiliary dense signals and specialized training procedures, such as KRLS, which improves dialog generation through reinforced keyword learning by incorporating dense rewards based on keyword relevance and dialog success metrics. These innovations collectively address the fundamental challenge of precise credit assignment in complex tasks, moving beyond sparse outcome-based rewards toward comprehensive process supervision.

Reward Bias Calibration. Reward models often exhibit systematic biases that misalign their predictions with true human preferences. These biases manifest as overconfidence in incorrect predictions, preferences for superficial characteristics like response length, and distributional misalignment between predicted and actual reward values. The core principle involves decomposing biased reward signals into true reward components and systematic bias terms, then applying corrective measures to improve alignment.

Our review of 19 papers reveals three primary approaches: *Post-Hoc Bias Correction* methods detect and remove bias after reward model training through mathematical decomposition and statistical correction, exemplified by [61] which decomposes biased rewards into true rewards plus bias terms using RC-Mean (estimating bias via local average rewards within neighborhoods) and RC-LWR (employing Locally Weighted Regression for robust bias estimation that adapts to

local data density). *Confidence-Based Calibration* leverages confidence scores and overconfidence detection to calibrate reward predictions during or after training, as demonstrated by PPO-M/PPO-C [83], where PPO-M calibrates reward models by augmenting training data with confidence-query prompts to prefer high confidence for correct responses, while PPO-C adjusts reward scores during PPO training using exponential moving averages and confidence-based calibration. *Distributional Methods* employ Bayesian inference, distributional modeling, and uncertainty quantification to handle reward model uncertainty and bias, such as the BIRL approach [9] that formulates LLM alignment as Bayesian Inverse Reinforcement Learning by learning posterior reward distributions $p(R|D)$ through variational inference. These strategies enable more reliable reward modeling that captures human judgment beyond surface-level characteristics.

Synthetic Preferences Generation. This refers to automatically generated preference signals that serve as alternatives or supplements to expensive human annotations, addressing the fundamental bottleneck in RLHF: the scarcity and cost of human preference data. The field has evolved beyond simple data augmentation to sophisticated methods that leverage diverse signal sources, iterative improvement, and cross-domain knowledge transfer.

We analyze 18 papers across six key innovation categories: *Cross-Modal Synthesis* leverages abundant preference data from one modality to improve reward models in data-scarce modalities, exemplified by RoVRM [170], which proposes multi-phase training to leverage textual preference abundance while maintaining multimodal capabilities for vision-language models. *Self-Evolutionary Loops* enable iterative self-improvement where reward models generate their own training data through progressive refinement cycles, as demonstrated by SER [59], which implements a curriculum-like self-improvement system with adaptive data filtering to achieve significant performance improvements over seed models. *Task-Performance Feedback* derives preference signals directly from downstream task performance rather than human annotations, with RLPF [187] using a frozen LLM to predict future user activities from generated summaries, where prediction accuracy serves as the reward signal, eliminating the need for human preference annotations. *Contrastive Automatic Labeling* automatically generates preference pairs through contrastive prompting or structural differences, as in RLCD [205]. *Knowledge-Guided Synthesis* employs external structured knowledge or domain expertise to guide preference generation, with OCEAN [188] leveraging knowledge graphs to create structured reasoning preferences. *Hybrid Multi-Component Synthesis* approaches like SALMON [161] combine multiple synthetic preference generation strategies, creating unified frameworks that can adapt to arbitrary human-defined principles without retraining while leveraging synthetic data to bootstrap initial preference understanding. These methods address human annotation scalability while maintaining alignment quality through automated feedback mechanisms.

Compositional Rewards. The systematic decomposition and combination of reward functions address the multi-faceted nature of LLM alignment. Unlike monolithic reward models that provide single scalar feedback, Compositional Reward systems break down desired behaviors into multiple components and combine them through various composition strategies. We categorize 21 papers into four primary methodological strategies.

Dynamic Weighting Optimization dynamically composes multiple reward functions with adaptive weighting mechanisms that adjust during training. Fast RL [85] treats reward composition as a max-min optimization problem, using mirror descent estimation to dynamically update weights in a weighted sum of multiple rewards, achieving superior performance across multiple metrics with improved stability compared to fixed weighting schemes. *Hierarchical Decomposition* decomposes complex reward functions into hierarchical structures with conditional activation. ALARM [78] decomposes reward modeling into two sub-tasks: (1) following holistic rewards until generation quality exceeds a threshold, then (2) combining holistic rewards with specific fine-grained rewards, creating a structure where general quality is prioritized before specific aspects are refined. *Structure Reward Engineering* explicitly encodes structural, syntactic, or rule-based constraints as compositional reward components. In VeriSeek [176], the reward function explicitly

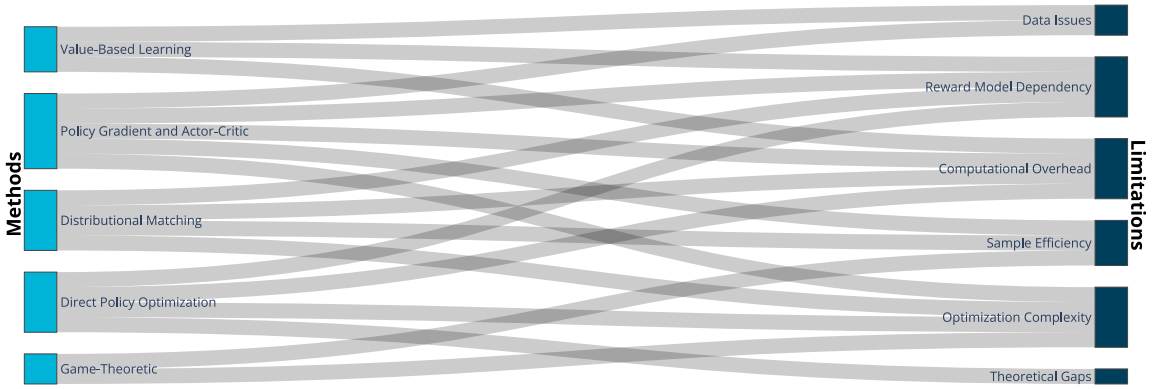

Fig. 6. Optimization Algorithm limitations: Sankey diagram showing distribution across five method categories.

decomposes code quality into syntactic validity, parseability, and structural similarity components. *Domain-Specific Decomposition* decomposes domain-specific quality metrics into multiple specialized reward components tailored to specific applications. LLM2ER-EQR [206] decomposes explanation quality into specialized components, including relevance, informativeness, persuasiveness, and user preference alignment, with each component measured through domain-specific metrics and combined into a composite reward function for PPO training. The field demonstrates a shift toward sophisticated reward engineering that moves beyond scalar rewards to better align with nuanced requirements.

RQ2: *What are the core methodological innovations within each method domain, and what patterns emerge in their technical contributions and practical applicability?*

Summary for RQ2: *The methodological innovations reveal distinct patterns across the three domains. Optimization Algorithms demonstrate how advanced methods build upon foundational algorithms like PPO and DPO, which serve as platforms for systematic enhancement and specialization, alongside development of other algorithms contributing to RL4LLM fine-tuning. Training Frameworks exhibit progressive patterns: modular decomposition, iterative self-improvement, hierarchical scalability, and real-time adaptation, reflecting a shift from predetermined frameworks to flexible, responsive architectures. Reward Modeling shows sophisticated patterns across the reward learning ecosystem, with innovations demonstrating holistic enhancement strategies that simultaneously address architectural design, signal quality, bias mitigation, data efficiency, and compositional complexity, shifting from monolithic reward functions toward multi-dimensional, engineered systems. These patterns collectively indicate the field's maturation toward principled and robust RL4LLM fine-tuning methodologies.*

3.3 Limitations and Future Directions (RQ3)

RQ3. What are the primary limitations across RL4LLM fine-tuning methods and what future work do researchers propose to address these limitations? Through systematic analysis of the extracted information for *C.1 Limitations* and *C.2 Future Directions* in Table 1, we comprehensively identify unresolved challenges, methodological gaps, and underexplored areas within each domain. This analysis maps the methodological innovations to their key limitations, providing a roadmap for future RL4LLM fine-tuning research.

Optimization Algorithm. These methods exhibit limitations including computational overhead, reward model dependency, and optimization complexity, as illustrated in Figure 6. Key limitations and future directions by category:

- *Data Issues* encompasses critical challenges where distribution shift in Policy Gradient and Actor-Critic leads to offline degradation and instability with larger models [79, 128], while Value-Based Learning exhibit limited online adaptation, with offline RL methodologies showing constrained transferability to online environments and only modest single-turn interaction improvements [194]. Research proposes four solution pathways: *Improving Distributional Robustness* through enhanced offline learning and alternative distance metrics [44, 79], *Bridging Offline-Online Learning Gaps* via real-time optimization adaptations [53], *Expanding LLM Action Spaces* for enhanced online learning capabilities [194], and *Establishing Dynamic Adaptation Paradigms* that transition from static preference datasets to online training where reward models co-evolve with policies, incorporating uncertainty estimates and enabling multi-step inference chains for multi-task generalization [27, 64, 77, 103].
- *Sample Efficiency* affects Policy Gradient and Actor-Critic methods, making them suffer from high polynomial sample complexity and 1.3× online sampling overhead [93, 222], while Game Theory algorithms lack theoretical assurances for finite-sample dynamics, relying on asymptotic foundations [228]. Distributional Matching methods exhibit biased log F estimation through sampling and noisy quantile estimation with limited samples, generating systematic Distributional Matching errors [1, 77, 146]. To address these issues, research proposes *Accelerating Sample Efficiency* through lookahead decoding and systematic prompt selection [45, 93, 222], and *Formulating Finite-Sample Theoretical Construction* to provide tangible performance boundaries [228].
- *Reward Model Dependency* affects multiple algorithm types with distinct vulnerabilities. Policy Gradient and Actor-Critic methods experience evaluation-optimization coupling requiring high-quality preference models [29, 41, 240], while Direct Policy Optimization methods show performance tightly coupled to reward model fidelity with over-optimization spawning "cheat patterns" [114, 119, 139, 179, 214]. Value-Based Learning algorithms become critically tied to reward function quality, introducing deployment fragility [47], and Distributional Matching performance fundamentally hinges on reward model fidelity for mode-seeking behaviors [1, 64, 103]. Research addresses these dependencies through *Enhancing Reward Modeling* via multi-objective frameworks and diversified AI feedback [41, 140, 240], *Pursuing Reward Model-Free Paradigms* using synthetic data generation [156, 214], and *Incorporating IRL-Extracted Rewards* into unified frameworks [47, 194].
- *Optimization Complexity* encompasses three challenges. Overoptimization tendency causes performance degradation when training beyond 40k samples in Policy Gradient and Actor-Critic methods [93, 128, 140] and reward hacking in Distributional Matching methods despite regularization [64]. Hyperparameter sensitivity affects Direct Policy Optimization requiring meticulous calibration without principled frameworks [57, 149, 224] and Game Theory methods relying on precise parameter tuning [175]. Convergence determination in Game Theory methods creates obstacles in establishing equilibrium states [175]. Research addresses these challenges through *Autonomous Policy Adjustment* that self-regulate reference policies and eliminate manual hyperparameter manipulation [175, 228], *Exploration Innovations* using adaptive divergence selection and off-policy integration [45, 115, 128], and *Convergence Detection Enhancement* for Nash Equilibrium identification [175].
- *Computational Overhead* affects multiple algorithm types. Policy Gradient and Actor-Critic methods require 3× space and 2× time versus standard fine-tuning [80, 140, 208], while Direct Policy Optimization methods demand GPU memory with approaches unvalidated beyond 7B parameters plus reward model training costs [65, 70, 107, 119, 179, 214]. Value-Based Learning requires double computational time versus supervised learning due to multiple transformer requirements [155], and Distributional Matching involves preprocessing costs scaling with sample size, extensive sample generation, and replay buffer maintenance [1, 27, 56, 103]. Research addresses these through *Scaling Breakthroughs* for larger architectures [70, 119, 179], *Streamlining Computational Architectures* by reducing transformer

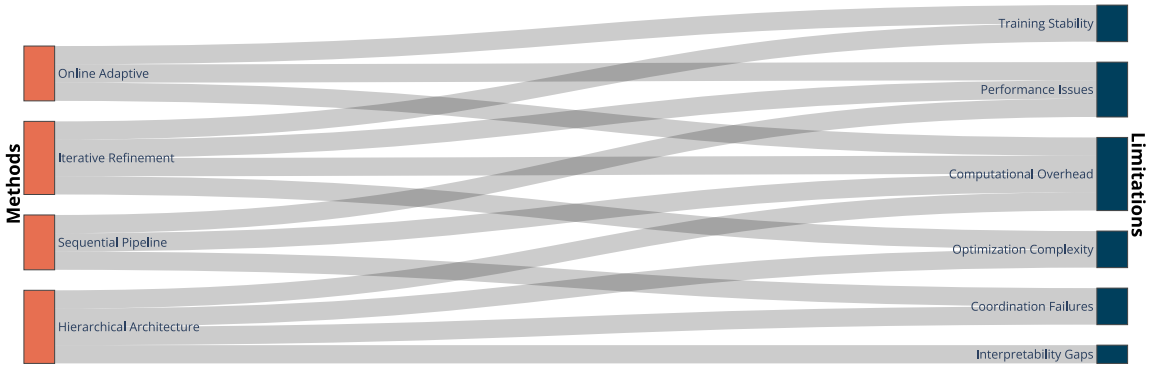

Fig. 7. Training Framework limitations: Sankey diagram showing distribution across four method categories.

dependencies [155], *Computational Efficiency* exploiting prompt similarity and learned approximations [1, 103], and *Advancing Architectural Scaling* beyond parameter constraints toward structured reasoning [27, 56, 64].

- *Theoretical Gaps* in Direct Policy Optimization methods arise as foundational DPO frameworks depend on Bradley-Terry preference models and break down under distributional shifts with broader or partially overlapping test distributions [57, 105]. Research addresses these through *Data-Driven Parameter Tuning* methodologies that eliminate manual exploration while proposing theoretical extensions beyond Bradley-Terry limitations [57], and *Broadening Theoretical Foundations* to accommodate dynamic distributional shifts [105].

Training Framework. These methods have extensive remaining limitations, including computational overhead in frameworks, persistent performance issues, and other specific limitations as illustrated in Figure 7. In the following, we elaborate on these limitations and future research directions by category in detail.

- *Performance Issues* manifest across three operational contexts. Sequential alignment flows exhibit $\sim$ 5% knowledge task degradation versus closed-source models due to alignment tax through pipeline stages, creating proprietary model dependency without systematic adversarial robustness testing [72, 90, 112, 184]. Online Adaptive frameworks suffer dual deterioration: off-policy learning degrades beyond 16 mini-batches creating continuous learning instability [127], while elevated inference latency impairs real-time responsiveness [202]. Iterative Refinement systems plateau after 2-3 iterations as model confidence and improvement gains deteriorate [18, 137, 210]. Future research addresses these through *Adaptive Sequential Flow Management* using dynamic token budgets and progress-based RL for denser reward signals [46, 90], *Asynchronous RLHF* enabling concurrent learning with advanced off-policy buffering [127], and *Real-Time Retrieval Architectures* minimizing latency via improved Progress Tracker designs and RLHF-trained neural reward models [137, 202].
- *Computational Overhead* manifests across all architectural paradigms. Sequential Pipelines are limited by 4K token contexts and single GPU training, preventing exploration of larger batches needed for pipeline depth. Resource constraints leave larger architectures unexplored [89, 144]. Iterative Refinement demands greater resources than single-pass methods, constraining practical iteration counts [137, 167]. Hierarchical Architectures increase training time by $\sim$ 20% through alternating hierarchy-level updates requiring multiple forward passes [164, 189]. Online Adaptive frameworks require complete parameter re-initialization at each step, rendering adaptation computationally prohibitive for deployment [122]. Future research pursues *Pipeline Scaling and Robustness* testing complete frameworks

on 34B/70B models while strengthening adversarial robustness [89, 90], *Trajectory-Based Learning Optimization* incorporating online RL into iterative self-training for reduced computational requirements [26], *Hierarchical Efficiency* reducing training overhead while preserving multi-level capabilities [189], and *Online Learning Integration* transitioning from costly re-initialization toward fine-tuning approaches leveraging recent data streams [122, 178].

- *Optimization Complexity* exhibits three key limitations. Hierarchical Architectures require experimental weight tuning between auxiliary and preference objectives with only moderate correlation between weights and evaluation metrics [3]. Iterative Refinement necessitates separate training phases rather than unified processes, restricting methods to 2-turn self-correction cycles [76, 137]. Current iterative systems optimize individual steps independently rather than complete trajectories, yielding suboptimal cumulative enhancements [55]. Future research pursues *Adaptive Architecture Exploration* across adapter designs for hierarchical tasks [67], *Unified Iterative Training* merging multi-stage refinement into seamless frameworks beyond current cycle barriers [18, 76], and *Joint Trajectory Optimization* of complete refinement sequences replacing greedy improvements [42, 55].
- *Training Stability* manifests in Online Adaptive frameworks that require KL coefficient adjustments across tasks, introducing instability during continuous learning [7]. Iterative Refinement frameworks depend on models' inherent self-refinement capabilities, which prove insufficient for complex reasoning [209, 225]. Future research pursues *Enhancing Self-Improvement Mechanisms* for Iterative Refinement beyond current approaches [209].
- *Coordination Failures* manifest in Sequential Pipelines that suffer domain-dependent RL effects where stages decline in simpler domains through overfitting, while the algorithm demonstrates highly unstable behavior during multi-stage fine-tuning, causing output degeneration and policy breakdown [46, 112, 144]. Hierarchical Architectures face dual challenges: hierarchical client grouping in FedBiscuit demands additional communication overhead while single selector approaches remain vulnerable to reward hacking [186]. Future research targets *End-to-End Pipeline Coordination* through unified sequential training and weakly-supervised alignment across retriever-to-generator pipelines [144, 184], *Federated Coordination Optimization* reducing communication overhead in hierarchical structures [186], *Multi-Feedback Aggregation* enabling hierarchical learning from diverse sources via sophisticated weight mechanisms [158], and *Ensemble Integration* employing multiple LLM agents with complementary strengths [31].
- *Interpretability Gaps* arise from Hierarchical Architectures where latent policies lack natural language interpretability, creating obstacles for understanding multi-level strategies [51]. Future research pursues *Adaptive Multi-Level Systems* through hierarchical explainability, enhancing interpretability across levels [51].

Reward Modeling. The reward-oriented methods encompass 9 extensive limitations. The connections between methods and limitations are shown in Figure 8. We analyze them below:

- *Data Issues* - Reward Modeling faces significant data-related challenges, particularly in Reward Granularity and Density methods where static reward models during policy training create distribution shifts between training and target distributions [177, 211]. These approaches also suffer from annotation noise, as automatic process supervision introduces false positives, especially with larger sampling values [177]. Solutions target multiple aspects: *Expanding Preference Simulation* beyond binary labels toward fine-grained multi-output rankings [205], *Implementing Iterative Training* paradigms for joint reward-policy evolution to address distribution shifts [177, 211], and *Establishing Hybrid Annotation* systems merging human expertise with automated processes for robust supervision [177]. Researchers also pursue *Developing Robust Autonomous Methods* for learning status identification and self-labeled data filtering [59], *Exploring Uncertainty-Guided Selection* using last-layer embeddings for active data selection [227], and *Implementing Federated Learning* approaches to unite reward models from disparate private datasets through weight averaging [141].

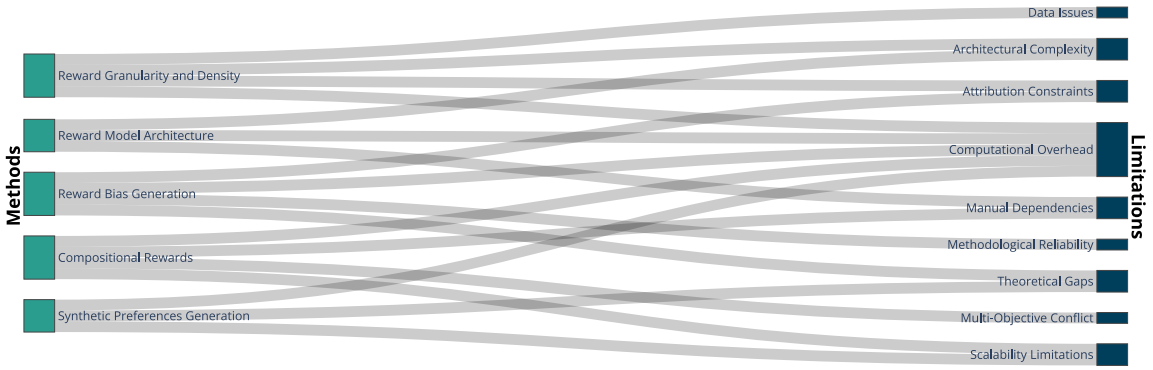

Fig. 8. Reward Modeling limitations: Sankey diagram showing distribution across five method categories.

- *Architectural Complexity* - Multiple design challenges plague Reward Model Architecture methods. Weight averaging approaches fail to harness diversity from heterogeneous architectures and cannot incorporate prediction disagreement for uncertainty estimation [141]. Complex frameworks like MoRE suffer elevated inference times due to adapter switching overhead. Tool-integrated architectures require significantly more training epochs to master invocations [88, 124]. Additionally, Reward Granularity and Density methods create tokenizer dependencies requiring identical tokenizers between reward and generative models [16]. To tackle these challenges, researchers pursue *Advancing Cross-Architecture Integration* through hybrid systems and sophisticated routers [124, 141]. Other efforts focus on *Enhancing Component Integration* for precise goal state derivation and Q-value disentanglement [125]. Additionally, *Strengthening Tool Integration* aims to improve adaptive invocation mechanisms [88, 106], while *Developing Cross-Tokenizer Mapping* explores alternative attribution methods [16].
- *Computational Overhead* - Reward Modeling methods impose substantial computational burdens across multiple dimensions. Advanced Reward Model Architectures, particularly MoE implementations, demand approximately 8× longer training periods, while multi-user architectures require maintaining multiple specialized models [15, 138]. Reward Granularity and Density methods dramatically increase GPU memory consumption and inference time through multiple reward model queries for fine-grained feedback [193]. Similarly, Reward Bias Calibration's ensemble-based preference estimation substantially elevates costs over single-model approaches [168]. PPO computational costs in Synthetic Preferences Generation prohibit comprehensive iterative experiments with LLM-in-the-loop approaches [59]. Compositional Rewards face the most severe constraints that multi-style approaches cannot explore beyond elementary combinations due to computational barriers, requiring elusive high-quality discriminators for data-scarce attributes, while multi-component gradient systems lack convergence guarantees [30, 120]. Solutions target *Optimizing Efficiency* by reducing 8× MoE overhead and streamlining MoRE frameworks [124, 138], alongside *Achieving Efficient Inference* through speculative decoding [193].
- *Attribution Constraints* - Reward Modeling methods struggle with comprehensive attribution across multiple dimensions. Reward Bias Calibration methods target individual characteristic biases rather than addressing multiple biases concurrently, limiting effectiveness in complex scenarios with intersecting biases [61]. Separately, Reward Granularity and Density approaches face distinct constraints. Attention-based methods remain restricted to positive-only token contributions while relying on questionable assumptions that attention mechanisms provide meaningful feature

attribution [16]. To address these limitations, researchers propose *Advancing Multi-Characteristic Calibration* systems for simultaneously handling multiple reward biases [61].

- *Scalability Limitations* - Reward Modeling methods also face scalability constraints across multiple dimensions. Synthetic Preferences Generation methods remain restricted to binary preference labels and specific task types, lacking nuanced modeling capabilities for scenarios [22, 205]. Compositional Rewards encounter constraints that reward decomposition methodologies cannot scale beyond moderate-scale models without billion-parameter validation, concentrating on narrow domains with ambiguous generalization potential. Multi-skill compositional alignment encompasses only rudimentary skill sets despite instructions demanding multiple capabilities [96, 117]. Research addresses these issues through *Amplifying Scalability Extensions* by implementing Compositional Rewards across billion-parameter models with diverse RL algorithms [117, 173] and *Orchestrating Comprehensive Multi-Skill Modeling* capable of handling multiple alignment capabilities per query [96].
- *Manual Dependencies* - Approaches suffer from extensive manual interventions that limit autonomy and scalability. Reward Model Architecture methods require predefined domain labels for routing decisions and manual capability dimension specifications for different tasks, preventing adaptive learning [124, 138]. Compositional Rewards face manual overhead: multi-objective approaches mandate task-specific hyperparameter calibration and manual constraint specification rather than autonomous learning. This dependence makes constraint range determination problematic while reducing generalization across diverse scenarios [232]. Researchers propose *Developing Automated Discovery* systems to replace manual capability definitions with intelligent discovery mechanisms [138], *Revolutionizing Automated Constraint Learning* to eliminate manual specification requirements [232], and *Developing Adaptive Calibration* mechanisms for automatic determination of calibration strength without manual hyperparameter tuning [61].
- *Multi-Objective Conflict* - Compositional Reward systems face inherent conflicts when integrating multiple reward signals, which degrade overall performance. As systems employ more reward models, they experience increasingly conflicting optimization signals, with three-reward configurations demonstrating deteriorated performance compared to two-reward setups. Performance typically declines after several epochs due to over-optimization effects, and multi-objective systems risk generating harmful content when component models exhibit biases or recognition failures [85, 173]. Solutions include *Advancing Multi-Objective Learning* to combine preference dimensions for comprehensive guidance by developing unified preference models that can handle multiple criteria simultaneously [211], *Architecting Saddle-Point Convergence* methods for multi-component systems that achieve stable equilibrium across competing objectives [120], *Materializing Robust Multi-Objective Frameworks* for managing beyond dual objectives while preserving reward-evaluator alignment through adaptive weighting mechanisms [173], and *Pioneering Conflict Resolution Mechanisms* through theoretical frameworks capable of identifying superfluous rewards and dynamically pruning redundant signals [85]. Future research focuses on developing hierarchical preference architectures and meta-learning approaches that automatically balance multiple reward components.
- *Methodological Reliability* - The Reward Bias Calibration approaches face fundamental challenges in their underlying assumptions. These methods depend on accurate confidence estimation, which becomes problematic with sparse or low-quality data that fails to provide sufficient samples in each forecast bin [58]. Reward models may also inadequately satisfy requirements for serving as unbiased instrumental variables, potentially introducing additional biases rather than eliminating them [197]. Solutions include *Leveraging Self-Improvement* approaches that utilize aligned confidence as supervisory signals [165] and *Implementing Unbiased Reward Modeling* methods ensuring reward models satisfy instrumental variable requirements [197].

- *Theoretical Gaps* - Current Reward Modeling methods suffer from insufficient theoretical foundations. Reward Bias Calibration methods necessitate strong assumptions including independence, sufficient density, and Lipschitz continuity, requiring manual calibration constant tuning when these assumptions fail [61]. Synthetic Preferences Generation approaches face different weaknesses by relying on empirical thresholds for data filtering strategies while lacking rigorous theoretical grounding for these critical design choices [59]. Researchers propose *Establishing Closed-Source Adaptation* techniques for extending calibration benefits beyond open-source models [165].

RQ3: *What are the primary limitations across RL4LLM fine-tuning methods and what future work do researchers propose to address these limitations?*

Summary for RQ3: *Current RL4LLM fine-tuning methods reveal systematic limitations across three primary domains while proposing diverse solution pathways. Computational overhead emerges as the universal constraint, affecting all method domains and highlighting a fundamental trade-off between task performance enhancement and computational efficiency. In Optimization Algorithms, methods encounter other frequently occurring limitations, including reward model dependency, sample efficiency, and optimization complexity, with future work addressing these algorithmic challenges. Training Frameworks present distinct limitations such as performance issues, coordination failures, training stability, and optimization complexity, highlighting that Training Framework research remains exploratory. Reward Modeling exhibits multiple reward-specific limitations, including manual dependencies, multi-objective conflicts, and scalability constraints, indicating substantial potential for RL4LLM fine-tuning improvement. The analysis reveals that while methods share some limitations, they pursue distinct solution strategies, indicating both the complexity of RL4LLM optimization and the active evolution toward more robust methodologies.*

4 Conclusion

In this comprehensive literature review, we focus on RL4LLM fine-tuning methods, where RL techniques are systematically applied to enhance LLM capabilities through parameter fine-tuning, specifically analyzing their methodological details. We categorize methods into three domains: Optimization Algorithm, Training Framework, and Reward Modeling, with further three-level hierarchies for subtle details. We examine 230 recent papers from 2022 to September 2025 using our research methodology to search relevant papers and extract important information. We establish research questions addressing 1) recent methods overview, 2) methodological innovations, and 3) limitations and future directions for in-depth analysis of RL4LLM fine-tuning methods.

Our findings reveal distinct patterns by method domain across target applications, challenges, and properties. Optimization Algorithm methods build upon foundational algorithms with mathematical enhancements and notably share remaining limitations: computational overhead, reward model dependency, optimization complexity, and sample efficiency. Training Frameworks can be divided into four main core frameworks, indicating a shift from predetermined to flexible and responsive architectures, promising for RL4LLM fine-tuning. These works target challenges in output quality and data efficiency while also having remaining limitations, including performance issues, coordination failures, training stability, and computational overhead. As a reward-oriented method domain, Reward Modeling has innovations that span different perspectives on architectural design, signal quality, bias mitigation, data efficiency, and compositional complexity, while highlighting extensive remaining limitations regarding reward-specific constraints. RL4LLM fine-tuning demonstrates significant progress through these various methods, while substantial limitations remain across all domains. Through systematic analysis of these methods and their innovations, this literature review provides researchers with essential guidance for advancing RL4LLM fine-tuning and identifying breakthrough research opportunities.

References

[1] Afra Amini, Tim Vieira, Elliott Ash, and Ryan Cotterell. Variational best-of-n alignment. In *ICLR*. OpenReview.net, 2025.

[2] Afra Amini, Tim Vieira, and Ryan Cotterell. Direct preference optimization with an offset. In *ACL (Findings)*, pages 9954–9972. Association for Computational Linguistics, 2024.

[3] Anirudhan Badrinath, Prabhat Agarwal, and Jiajing Xu. Unified preference optimization: Language model alignment beyond the preference frontier. *Trans. Mach. Learn. Res.*, 2025, 2025.

[4] Ashutosh Baheti, Ximing Lu, Faeze Brahman, Ronan Le Bras, Maarten Sap, and Mark O. Riedl. Leftover lunch: Advantage-based offline reinforcement learning for language models. In *ICLR*. OpenReview.net, 2024.

[5] Chenjia Bai, Yang Zhang, Shuang Qiu, Qiaosheng Zhang, Kang Xu, and Xuelong Li. Online preference alignment for language models via count-based exploration. In *ICLR*. OpenReview.net, 2025.

[6] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Chris Olah, Benjamin Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862, 2022.

[7] Liat Bezalel, Eyal Orgad, and Amir Globerson. Teaching models to improve on tape. In *AAAI*, pages 15550–15558. AAAI Press, 2025.

[8] Alexander Bukharin, Ilgee Hong, Haoming Jiang, Zichong Li, Qingru Zhang, Zixuan Zhang, and Tuo Zhao. Robust reinforcement learning from corrupted human feedback. In *NeurIPS*, 2024.

[9] Yuang Cai, Yuyu Yuan, Jinsheng Shi, and Qinhong Lin. Approximated variational bayesian inverse reinforcement learning for large language model alignment. In *AAAI*, pages 23505–23513. AAAI Press, 2025.

[10] Daniele Calandriello, Zhaohan Daniel Guo, Rémi Munos, Mark Rowland, Yunhao Tang, Bernardo Ávila Pires, Pierre Harvey Richemond, Charline Le Lan, Michal Valko, Tianqi Liu, Rishabh Joshi, Zeyu Zheng, and Bilal Piot. Human alignment of large language models through online preference optimisation. In *ICML*. OpenReview.net, 2024.

[11] Meng Cao, Yue Dong, and Jackie Chi Kit Cheung. Hallucinated but factual! inspecting the factuality of hallucinations in abstractive summarization. In *ACL (1)*, pages 3340–3354. Association for Computational Linguistics, 2022.

[12] Meng Cao, Lei Shu, Lei Yu, Yun Zhu, Nevan Wichers, Yinxiao Liu, and Lei Meng. Enhancing reinforcement learning with dense rewards from language model critic. In *EMNLP*, pages 9119–9138. Association for Computational Linguistics, 2024.

[13] Shicong Cen, Jincheng Mei, Katayoon Goshvadi, Hanjun Dai, Tong Yang, Sherry Yang, Dale Schuurmans, Yuejie Chi, and Bo Dai. Value-incentivized preference optimization: A unified approach to online and offline RLHF. In *ICLR*. OpenReview.net, 2025.

[14] Yekun Chai, Haoran Sun, Huang Fang, Shuohuan Wang, Yu Sun, and Hua Wu. MA-RLHF: reinforcement learning from human feedback with macro actions. In *ICLR*. OpenReview.net, 2025.

[15] Souradip Chakraborty, Jiahao Qiu, Hui Yuan, Alec Koppel, Dinesh Manocha, Furong Huang, Amrit S. Bedi, and Mengdi Wang. Maxmin-rlhf: Alignment with diverse human preferences. In *ICML*. OpenReview.net, 2024.

[16] Alex James Chan, Hao Sun, Samuel Holt, and Mihaela van der Schaar. Dense reward for free in reinforcement learning from human feedback. In *ICML*. OpenReview.net, 2024.

[17] Sapana Chaudhary, Ujwal Dinesha, Dileep Kalathil, and Srinivas Shakkottai. Risk-averse fine-tuning of large language models. In *NeurIPS*, 2024.

[18] Changyu Chen, Zichen Liu, Chao Du, Tianyu Pang, Qian Liu, Arunesh Sinha, Pradeep Varakantham, and Min Lin. Bootstrapping language models with DPO implicit rewards. In *ICLR*. OpenReview.net, 2025.

[19] Guoxin Chen, Minpeng Liao, Chengxi Li, and Kai Fan. Step-level value preference optimization for mathematical reasoning. In *EMNLP (Findings)*, pages 7889–7903. Association for Computational Linguistics, 2024.

[20] Haoxian Chen, Hanyang Zhao, Henry Lam, David D. Yao, and Wenpin Tang. Mallowspo: Fine-tune your LLM with preference dispersions. In *ICLR*. OpenReview.net, 2025.

[21] Lichang Chen, Chen Zhu, Jiuhai Chen, Davit Soselia, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. ODIN: disentangled reward mitigates hacking in RLHF. In *ICML*. OpenReview.net, 2024.

[22] Lu Chen, Rui Zheng, Binghai Wang, Senjie Jin, Caishuang Huang, Junjie Ye, Zhihao Zhang, Yuhao Zhou, Zhiheng Xi, Tao Gui, Qi Zhang, and Xuanjing Huang. Improving discriminative capability of reward models in RLHF using contrastive learning. In *EMNLP*, pages 15270–15283. Association for Computational Linguistics, 2024.

[23] Yangyi Chen, Karan Sikka, Michael Cogswell, Heng Ji, and Ajay Divakaran. DRESS : Instructing large vision-language models to align and interact with humans via natural language feedback. In *CVPR*, pages 14239–14250. IEEE, 2024.

[24] Zhaorun Chen, Zhuokai Zhao, Zhihong Zhu, Ruiqi Zhang, Xiang Li, Bhiksha Raj, and Huaxiu Yao. Autoprm: Automating procedural supervision for multi-step reasoning via controllable question decomposition. In *NAACL-HLT*, pages 1346–1362. Association for Computational Linguistics, 2024.

[25] Zhipeng Chen, Kun Zhou, Xin Zhao, Junchen Wan, Fuzheng Zhang, Di Zhang, and Ji-Rong Wen. Improving large language models via fine-grained reinforcement learning with minimum editing constraint. In *ACL (Findings)*, pages 5694–5711. Association for Computational Linguistics, 2024.

[26] Pengyu Cheng, Tianhao Hu, Han Xu, Zhisong Zhang, Yong Dai, Lei Han, Nan Du, and Xiaolong Li. Self-playing adversarial language game enhances LLM reasoning. In *NeurIPS*, 2024.

[27] Yinlam Chow, Guy Tennenholtz, Izzeddin Gur, Vincent Zhuang, Bo Dai, Aviral Kumar, Rishabh Agarwal, Sridhar Thiagarajan, Craig Boutilier, and Aleksandra Faust. Inference-aware fine-tuning for best-of-n sampling in large language models. In *ICLR*. OpenReview.net, 2025.

[28] Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: safe reinforcement learning from human feedback. In *ICLR*. OpenReview.net, 2024.

[29] Juntao Dai, Taiye Chen, Yaodong Yang, Qian Zheng, and Gang Pan. Mitigating reward over-optimization in RLHF via behavior-supported regularization. In *ICLR*. OpenReview.net, 2025.

[30] Karin de Langis, Ryan Koo, and Dongyeop Kang. Dynamic multi-reward weighting for multi-style controllable generation. In *EMNLP*, pages 6783–6800. Association for Computational Linguistics, 2024.

[31] Yang Deng, Wenxuan Zhang, Wai Lam, See-Kiong Ng, and Tat-Seng Chua. Plug-and-play policy planner for large language model powered dialogue agents. In *ICLR*. OpenReview.net, 2024.

[32] Guanting Dong, Keming Lu, Chengpeng Li, Tingyu Xia, Bowen Yu, Chang Zhou, and Jingren Zhou. Self-play with execution feedback: Improving instruction-following capabilities of large language models. In *ICLR*. OpenReview.net, 2025.

[33] Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. RAFT: reward ranked finetuning for generative foundation model alignment. *Trans. Mach. Learn. Res.*, 2023, 2023.

[34] Qian Dong, Yiding Liu, Qingyao Ai, Zhijing Wu, Haitao Li, Yiqun Liu, Shuaiqiang Wang, Dawei Yin, and Shaoping Ma. Unsupervised large language model alignment for information retrieval via contrastive feedback. In *SIGIR*, pages 48–58. ACM, 2024.

[35] Shihan Dou, Yan Liu, Haoxiang Jia, Enyu Zhou, Limao Xiong, Junjie Shan, Caishuang Huang, Xiao Wang, Xiaoran Fan, Zhiheng Xi, Yuhao Zhou, Tao Ji, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Stepcoder: Improving code generation with reinforcement learning from compiler feedback. In *ACL (1)*, pages 4571–4585. Association for Computational Linguistics, 2024.

[36] Subhabrata Dutta, Ishan Pandey, Joykirat Singh, Sunny Manchanda, Soumen Chakrabarti, and Tanmoy Chakraborty. Frugal lms trained to invoke symbolic solvers achieve parameter-efficient arithmetic reasoning. In *AAAI*, pages 17951–17959. AAAI Press, 2024.

[37] Farshid Faal, Ketra A. Schmitt, and Jia Yuan Yu. Reward modeling for mitigating toxicity in transformer-based language models. *Appl. Intell.*, 53(7):8421–8435, 2023.

[38] Flint Xiaofeng Fan, Cheston Tan, Yew-Soon Ong, Roger Wattenhofer, and Wei Tsang Ooi. Fedrlhf: A convergence-guaranteed federated framework for privacy-preserving and personalized RLHF. In *AAMAS*, pages 713–721. International Foundation for Autonomous Agents and Multiagent Systems / ACM, 2025.

[39] Jianzhou Feng, Qin Wang, Huaxiao Qiu, and Lirong Liu. Retrieval in decoder benefits generative models for explainable complex question answering. *Neural Networks*, 181:106833, 2025.

[40] Yunlong Feng, Yang Xu, Libo Qin, Yasheng Wang, and Wanxiang Che. Improving language model reasoning with self-motivated learning. In *LREC/COLING*, pages 8840–8852. ELRA and ICCL, 2024.

[41] Yannis Flet-Berliac, Nathan Grinsztajn, Florian Strub, Eugene Choi, Bill Wu, Chris Cremer, Arash Ahmadian, Yash Chandak, Mohammad Gheshlaghi Azar, Olivier Pietquin, and Matthieu Geist. Contrastive policy gradient: Aligning llms on sequence-level scores in a supervised-friendly fashion. In *EMNLP*, pages 21353–21370. Association for Computational Linguistics, 2024.

[42] Dylan J. Foster, Zakaria Mhammedi, and Dhruv Rohatgi. Is a good foundation necessary for efficient reinforcement learning? the computational role of the base model in exploration. In *COLT*, volume 291 of *Proceedings of Machine Learning Research*, pages 2026–2142. PMLR, 2025.

[43] Zhaolin Gao, Jonathan D. Chang, Wenhao Zhan, Owen Oertell, Gokul Swamy, Kianté Brantley, Thorsten Joachims, Drew Bagnell, Jason D. Lee, and Wen Sun. REBEL: reinforcement learning via regressing relative rewards. In *NeurIPS*, 2024.

[44] Zhaolin Gao, Wenhao Zhan, Jonathan Daniel Chang, Gokul Swamy, Kianté Brantley, Jason D. Lee, and Wen Sun. Regressing the relative future: Efficient policy optimization for multi-turn RLHF. In *ICLR*. OpenReview.net, 2025.

[45] Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, Nahyeon Ryu, and Marc Dymetman. Aligning language models with preferences through f-divergence minimization. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 11546–11583. PMLR, 2023.

[46] Shangding Gu, Alois Knoll, and Ming Jin. Teams-rl: Teaching llms to generate better instruction datasets via reinforcement learning. *Trans. Mach. Learn. Res.*, 2024, 2024.

[47] Han Guo, Bowen Tan, Zhengzhong Liu, Eric P. Xing, and Zhiting Hu. Efficient (soft) q-learning for text generation with limited good data. In *EMNLP (Findings)*, pages 6969–6991. Association for Computational Linguistics, 2022.

[48] Jiuzhou Han, Wray L. Buntine, and Ehsan Shareghi. Reward engineering for generating semi-structured explanation. In *EACL (Findings)*, pages 589–602. Association for Computational Linguistics, 2024.

[49] Shugang Hao and Lingjie Duan. Online learning from strategic human feedback in LLM fine-tuning. In *ICASSP*, pages 1–5. IEEE, 2025.

[50] Alexander Havrilla, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, and Roberta Raileanu. Glore: When, where, and how to improve LLM reasoning via global and local refinements. In *ICML*. OpenReview.net, 2024.

[51] Tao He, Lizi Liao, Yixin Cao, Yuanxing Liu, Yiheng Sun, Zerui Chen, Ming Liu, and Bing Qin. Simulation-free hierarchical latent policy planning for proactive dialogues. In *AAAI*, pages 24032–24040. AAAI Press, 2025.

[52] Ilgee Hong, Zichong Li, Alexander Bukharin, Yixiao Li, Haoming Jiang, Tianbao Yang, and Tuo Zhao. Adaptive preference scaling for reinforcement learning with human feedback. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng

Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

[53] Joey Hong, Anca D. Dragan, and Sergey Levine. Q-SFT: q-learning for language models via supervised fine-tuning. In *ICLR*. OpenReview.net, 2025.

[54] Zhang-Wei Hong, Idan Shenfeld, Tsun-Hsuan Wang, Yung-Sung Chuang, Aldo Pareja, James R. Glass, Akash Srivastava, and Pulkit Agrawal. Curiosity-driven red-teaming for large language models. In *ICLR*. OpenReview.net, 2024.

[55] Sheryl Hsu, Omar Khattab, Chelsea Finn, and Archit Sharma. Grounding by trying: Llms with reinforcement learning-enhanced retrieval. In *ICLR*. OpenReview.net, 2025.

[56] Edward J. Hu, Moksh Jain, Eric Elmoznino, Younesse Kaddar, Guillaume Lajoie, Yoshua Bengio, and Nikolay Malkin. Amortizing intractable inference in large language models. In *ICLR*. OpenReview.net, 2024.

[57] Audrey Huang, Wenhao Zhan, Tengyang Xie, Jason D. Lee, Wen Sun, Akshay Krishnamurthy, and Dylan J. Foster. Correcting the mythos of kl-regularization: Direct alignment without overoptimization via chi-squared preference optimization. In *ICLR*. OpenReview.net, 2025.

[58] Baihe Huang, Hiteshi Sharma, and Yi Mao. Enhancing language model alignment: A confidence-based approach to label smoothing. In *EMNLP*, pages 21341–21352. Association for Computational Linguistics, 2024.

[59] Chenghua Huang, Zhizhen Fan, Lu Wang, Fangkai Yang, Pu Zhao, Zeqi Lin, Qingwei Lin, Dongmei Zhang, Saravan Rajmohan, and Qi Zhang. Self-evolved reward learning for LLMS. In *ICLR*. OpenReview.net, 2025.

[60] Xinmeng Huang, Shuo Li, Edgar Dobriban, Osbert Bastani, Hamed Hassani, and Dongsheng Ding. One-shot safety alignment for large language models via optimal dualization. In *NeurIPS*, 2024.

[61] Zeyu Huang, Zihan Qiu, Zili Wang, Edoardo M. Ponti, and Ivan Titov. Post-hoc reward calibration: A case study on length bias. In *ICLR*. OpenReview.net, 2025.

[62] Youngsoo Jang, Geon-Hyeong Kim, Byoungjip Kim, Yu Jin Kim, Honglak Lee, and Moontae Lee. Degeneration-free policy optimization: RL fine-tuning for language models without degeneration. In *ICML*. OpenReview.net, 2024.

[63] Youngsoo Jang, Jongmin Lee, and Kee-Eung Kim. Gpt-critic: Offline reinforcement learning for end-to-end task-oriented dialogue systems. In *ICLR*. OpenReview.net, 2022.

[64] Haozhe Ji, Cheng Lu, Yilin Niu, Pei Ke, Hongning Wang, Jun Zhu, Jie Tang, and Minlie Huang. Towards efficient exact optimization of language model alignment. In *ICML*. OpenReview.net, 2024.

[65] Kaixuan Ji, Jiafan He, and Quanquan Gu. Reinforcement learning from human feedback with active queries. *Trans. Mach. Learn. Res.*, 2025, 2025.

[66] Liqiang Jing and Xinya Du. FGAIF: aligning large vision-language models with fine-grained AI feedback. *Trans. Mach. Learn. Res.*, 2025, 2025.

[67] DaeJin Jo, Taehwan Kwon, Eun-Sol Kim, and Sungwoong Kim. Selective token generation for few-shot natural language generation. In *COLING*, pages 5837–5856. International Committee on Computational Linguistics, 2022.

[68] Nurgali Kadyrbek, Zhanseit Tuimebayev, Madina Mansurova, and Vítor Viegas. The development of small-scale language models for low-resource languages, with a focus on kazakh and direct preference optimization. *Big Data Cogn. Comput.*, 9(5):137, 2025.

[69] Aly M. Kassem, Omar Mahmoud, and Sherif Saad. Preserving privacy through dememorization: An unlearning technique for mitigating memorization risks in language models. In *EMNLP*, pages 4360–4379. Association for Computational Linguistics, 2023.

[70] Saeed Khaki, JinJin Li, Lan Ma, Liu Yang, and Prathap Ramachandra. RS-DPO: A hybrid rejection sampling and direct preference optimization method for alignment of large language models. In *NAACL-HLT (Findings)*, pages 1665–1680. Association for Computational Linguistics, 2024.

[71] Kyuyoung Kim, Ah Jeong Seo, Hao Liu, Jinwoo Shin, and Kimin Lee. Margin matching preference optimization: Enhanced model alignment with granular feedback. In *EMNLP (Findings)*, pages 13554–13570. Association for Computational Linguistics, 2024.

[72] Sungdong Kim, Sanghwan Bae, Jamin Shin, Soyoung Kang, Donghyun Kwak, Kang Min Yoo, and Minjoon Seo. Aligning large language models through synthetic feedback. In *EMNLP*, pages 13677–13700. Association for Computational Linguistics, 2023.

[73] Barbara Kitchenham et al. Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004):1–26, 2004.

[74] Lingxiao Kong, Cong Yang, Susanne Neufang, Oya Deniz Beyan, and Zeyd Boukhers. EMORL: ensemble multi-objective reinforcement learning for efficient and flexible LLM fine-tuning. *CoRR*, abs/2505.02579, 2025.

[75] Weize Kong, Spurthi Amba Hombaiah, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. Prewrite: Prompt rewriting with reinforcement learning. In *ACL (Short Papers)*, pages 594–601. Association for Computational Linguistics, 2024.

[76] Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D. Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei M. Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal M. P. Behbahani, and Aleksandra Faust. Training language models to self-correct via reinforcement learning. In *ICLR*. OpenReview.net, 2025.

[77] Oh Joon Kwon, Daiki E. Matsunaga, and Kee-Eung Kim. GDPO: learning to directly align language models with diversity using gflownets. In *EMNLP*, pages 17120–17139. Association for Computational Linguistics, 2024.

[78] Yuhang Lai, Siyuan Wang, Shujun Liu, Xuanjing Huang, and Zhongyu Wei. Alarm: Align language models via hierarchical rewards modeling. In *ACL (Findings)*, pages 7817–7831. Association for Computational Linguistics, 2024.

[79] Hao Lang, Fei Huang, and Yongbin Li. Fine-tuning language models with reward learning on policy. In *NAACL-HLT*, pages 1382–1392. Association for Computational Linguistics, 2024.

[80] Hung Le, Yue Wang, Akhilesh Deepak Gotmare, Silvio Savarese, and Steven Chu-Hong Hoi. Coderl: Mastering code generation through pretrained models and deep reinforcement learning. In *NeurIPS*, 2022.

[81] Changhun Lee and Chiehyeon Lim. Towards pareto-efficient RLHF: paying attention to a few high-reward samples with reward dropout. In *EMNLP (Findings)*, pages 8335–8349. Association for Computational Linguistics, 2024.

[82] Janghwan Lee, Seongmin Park, Sukjin Hong, Minsoo Kim, Du-Seong Chang, and Jungwook Choi. Improving conversational abilities of quantized large language models via direct preference alignment. In *ACL (1)*, pages 11346–11364. Association for Computational Linguistics, 2024.

[83] Jixuan Leng, Chengsong Huang, Banghua Zhu, and Jiaxin Huang. Taming overconfidence in llms: Reward calibration in RLHF. In *ICLR*. OpenReview.net, 2025.

[84] Dongyang Li, Taolin Zhang, Longtao Huang, Chengyu Wang, Xiaofeng He, and Hui Xue. KEHRL: learning knowledge-enhanced language representations with hierarchical reinforcement learning. In *LREC/COLING*, pages 9693–9704. ELRA and ICCL, 2024.

[85] Jiahui Li, Hanlin Zhang, Fengda Zhang, Tai-Wei Chang, Kun Kuang, Long Chen, and Jun Zhou. Optimizing language models with fair and stable reward composition in reinforcement learning. In *EMNLP*, pages 10122–10140. Association for Computational Linguistics, 2024.

[86] Jian Li, Haojing Huang, Yujia Zhang, Pengfei Xu, Xi Chen, Rui Song, Lida Shi, Jingwen Wang, and Hao Xu. Self-supervised preference optimization: Enhance your language model with preference degree awareness. In *EMNLP (Findings)*, pages 14452–14466. Association for Computational Linguistics, 2024.

[87] Jiaxiang Li, Siliang Zeng, Hoi-To Wai, Chenliang Li, Alfredo García, and Mingyi Hong. Getting more juice out of the SFT data: Reward learning from human demonstration improves SFT for LLM alignment. In *NeurIPS*, 2024.

[88] Lei Li, Yekun Chai, Shuohuan Wang, Yu Sun, Hao Tian, Ningyu Zhang, and Hua Wu. Tool-augmented reward modeling. In *ICLR*. OpenReview.net, 2024.

[89] Long Li, Xuzheng He, Haozhe Wang, Linlin Wang, and Liang He. How do humans write code? large models do it the same way too. In *EMNLP*, pages 4638–4649. Association for Computational Linguistics, 2024.

[90] Sha Li and Naren Ramakrishnan. Oreo: A plug-in context reconstructor to enhance retrieval-augmented generation. *CoRR*, abs/2502.13019, 2025.

[91] Wendi Li, Wei Wei, Kaihe Xu, Wenfeng Xie, Dangyang Chen, and Yu Cheng. Reinforcement learning with token-level feedback for controllable text generation. In *NAACL-HLT (Findings)*, pages 1704–1719. Association for Computational Linguistics, 2024.

[92] Yi-Chen Li, Fuxiang Zhang, Wenjie Qiu, Lei Yuan, Chengxing Jia, Zongzhang Zhang, Yang Yu, and Bo An. Q-adapter: Customizing pre-trained llms to new preferences with forgetting mitigation. In *ICLR*. OpenReview.net, 2025.

[93] Ziniu Li, Tian Xu, Yushun Zhang, Zhihang Lin, Yang Yu, Ruoyu Sun, and Zhi-Quan Luo. Remax: A simple, effective, and efficient reinforcement learning method for aligning large language models. In *ICML*. OpenReview.net, 2024.

[94] Yuxin Liang, Zhuoyang Song, Hao Wang, and Jiaxing Zhang. Learning to trust your feelings: Leveraging self-awareness in llms for hallucination mitigation. *CoRR*, abs/2401.15449, 2024.

[95] Kuo Liao, Shuang Li, Meng Zhao, Liqun Liu, Mengge Xue, Zhenyu Hu, Honglin Han, and Chengguo Yin. Enhancing reinforcement learning with label-sensitive reward for natural language understanding. In *ACL (1)*, pages 4206–4220. Association for Computational Linguistics, 2024.

[96] Sheng-Chieh Lin, Luyu Gao, Barlas Oguz, Wenhan Xiong, Jimmy Lin, Scott Yih, and Xilun Chen. FLAME : Factuality-aware alignment for large language models. In *NeurIPS*, 2024.

[97] Tzu-Han Lin, Chen-An Li, Hung-yi Lee, and Yun-Nung Chen. Dogerm: Equipping reward models with domain knowledge through model merging. In *EMNLP*, pages 15506–15524. Association for Computational Linguistics, 2024.

[98] Jiacheng Liu, Skyler Hallinan, Ximing Lu, Pengfei He, Sean Welleck, Hannaneh Hajishirzi, and Yejin Choi. Rainier: Reinforced knowledge introspector for commonsense question answering. In *EMNLP*, pages 8938–8958. Association for Computational Linguistics, 2022.

[99] Keliang Liu, Dingkang Yang, Ziyun Qian, Weijie Yin, Yuchi Wang, Hongsheng Li, Jun Liu, Peng Zhai, Yang Liu, and Lihua Zhang. Reinforcement learning meets large language models: A survey of advancements and applications across the LLM lifecycle. *CoRR*, abs/2509.16679, 2025.

[100] Renpu Liu, Peng Wang, Donghao Li, Cong Shen, and Jing Yang. A shared low-rank adaptation approach to personalized RLHF. In *AISTATS*, volume 258 of *Proceedings of Machine Learning Research*, pages 1405–1413. PMLR, 2025.

[101] Ruibo Liu, Chenyan Jia, Ge Zhang, Ziyu Zhuang, Tony X. Liu, and Soroush Vosoughi. Second thoughts are best: Learning to re-align with human values from text edits. In *NeurIPS*, 2022.

[102] Ruibo Liu, Ge Zhang, Xinyu Feng, and Soroush Vosoughi. Aligning generative language models with human values. In *NAACL-HLT (Findings)*, pages 241–252. Association for Computational Linguistics, 2022.

[103] Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J. Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. In *ICLR*. OpenReview.net, 2024.

[104] Saüc Abadal Lloret, Shehzaad Dhuliawala, Keerthiram Murugesan, and Mrinmaya Sachan. Towards aligning language models with textual feedback. In *EMNLP*, pages 20240–20266. Association for Computational Linguistics, 2024.

[105] Thanawat Lodkaew, Tongtong Fang, Takashi Ishida, and Masashi Sugiyama. Importance weighting for aligning language models under deployment distribution shift. *Trans. Mach. Learn. Res.*, 2025, 2025.

[106] Ángela López-Cardona, Carlos Segura, Alexandros Karatzoglou, Sergi Abadal, and Ioannis Arapakis. Seeing eye to AI: human alignment via gaze-based response rewards for large language models. In *ICLR*. OpenReview.net, 2025.

[107] Junru Lu, Jiazheng Li, Siyu An, Meng Zhao, Yulan He, Di Yin, and Xing Sun. Eliminating biased length reliance of direct preference optimization via down-sampled KL divergence. In *EMNLP*, pages 1047–1067. Association for Computational Linguistics, 2024.

[108] Ximing Lu, Sean Welleck, Jack Hessel, Liwei Jiang, Lianhui Qin, Peter West, Prithviraj Ammanabrolu, and Yejin Choi. QUARK: controllable text generation with reinforced unlearning. In *NeurIPS*, 2022.

[109] Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Qingwei Lin, Jian-Guang Lou, Shifeng Chen, Yansong Tang, and Weizhu Chen. Wizardarena: Post-training large language models via simulated offline chatbot arena. In *NeurIPS*, 2024.

[110] Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jian-Guang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, Yansong Tang, and Dongmei Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. In *ICLR*. OpenReview.net, 2025.

[111] Hao Ma, Tianyi Hu, Zhiqiang Pu, Boyin Liu, Xiaolin Ai, Yanyan Liang, and Min Chen. Coevolving with the other you: Fine-tuning LLM with sequential cooperative multi-agent reinforcement learning. In *NeurIPS*, 2024.

[112] Ali Malik, Stephen Mayhew, Christopher Piech, and Klinton Bicknell. From tarzan to tolkien: Controlling the language proficiency level of llms for content generation. In *ACL (Findings)*, pages 15670–15693. Association for Computational Linguistics, 2024.

[113] Shengyu Mao, Yong Jiang, Boli Chen, Xiao Li, Peng Wang, Xinyu Wang, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. Rafe: Ranking feedback improves query rewriting for RAG. In *EMNLP (Findings)*, pages 884–901. Association for Computational Linguistics, 2024.

[114] Xin Mao, Feng-Lin Li, Huimin Xu, Wei Zhang, Wang Chen, and Anh Tuan Luu. Don't forget your reward values: Language model alignment via value-based calibration. In *EMNLP*, pages 17622–17642. Association for Computational Linguistics, 2024.

[115] Alice Martin, Guillaume Quispe, Charles Ollion, Sylvain Le Corff, Florian Strub, and Olivier Pietquin. Learning natural language generation with truncated reinforcement learning. In *NAACL-HLT*, pages 12–37. Association for Computational Linguistics, 2022.

[116] Yuchun Miao, Sen Zhang, Liang Ding, Rong Bao, Lefei Zhang, and Dacheng Tao. Inform: Mitigating reward hacking in RLHF via information-theoretic reward modeling. In *NeurIPS*, 2024.

[117] Do June Min, Verónica Pérez-Rosas, Ken Resnicow, and Rada Mihalcea. Dynamic reward adjustment in multi-reward reinforcement learning for counselor reflection generation. In *LREC/COLING*, pages 5437–5449. ELRA and ICCL, 2024.

[118] David Moher, Alessandro Liberati, Jennifer Tetzlaff, Douglas G Altman, Prisma Group, et al. Preferred reporting items for systematic reviews and meta-analyses: the prisma statement. *International journal of surgery*, 8(5):336–341, 2010.

[119] Tetsuro Morimura, Mitsuki Sakamoto, Yuu Jinnai, Kenshi Abe, and Kaito Ariu. Filtered direct preference optimization. In *EMNLP*, pages 22729–22770. Association for Computational Linguistics, 2024.

[120] Ted Moskovitz, Aaditya K. Singh, DJ Strouse, Tuomas Sandholm, Ruslan Salakhutdinov, Anca D. Dragan, and Stephen Marcus McAleer. Confronting reward model overoptimization with constrained RLHF. In *ICLR*. OpenReview.net, 2024.

[121] Tong Mu, Alec Helyar, Johannes Heidecke, Joshua Achiam, Andrea Vallone, Ian Kivlichan, Molly Lin, Alex Beutel, John Schulman, and Lilian Weng. Rule based rewards for language model safety. In *NeurIPS*, 2024.

[122] William Muldrew, Peter Hayes, Mingtian Zhang, and David Barber. Active preference learning for large language models. In *ICML*. OpenReview.net, 2024.

[123] Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Côme Fiegel, Andrea Michi, Marco Selvi, Sertan Girgin, Nikola Momchev, Olivier Bachem, Daniel J. Mankowitz, Doina Precup, and Bilal Piot. Nash learning from human feedback. In *ICML*. OpenReview.net, 2024.

[124] Hyuk Namgoong, Jeesu Jung, Sangkeun Jung, and Yoon-Hyung Roh. Exploring domain robust lightweight reward models based on router mechanism. In *ACL (Findings)*, pages 8644–8652. Association for Computational Linguistics, 2024.

[125] Vaskar Nath, Dylan Slack, Jeff Da, Yuntao Ma, Hugh Zhang, Spencer Whitehead, and Sean Hendryx. Learning goal-conditioned representations for language reward models. In *NeurIPS*, 2024.

[126] Minh Nguyen, Toàn Quoc Nguyên, Kishan KC, Zeyu Zhang, and Thuy Vu. Reinforcement learning from answer reranking feedback for retrieval-augmented answer generation. In *INTERSPEECH*. ISCA, 2024.

[127] Michael Noukhovitch, Shengyi Huang, Sophie Xhonneux, Arian Hosseini, Rishabh Agarwal, and Aaron C. Courville. Asynchronous RLHF: faster and more efficient off-policy RL for language models. In *ICLR*. OpenReview.net, 2025.

[128] Michael Noukhovitch, Samuel Lavoie, Florian Strub, and Aaron C. Courville. Language model alignment with elastic reset. In *NeurIPS*, 2023.

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

[130] Gaurav Pandey, Yatin Nandwani, Tahira Naseem, Mayank Mishra, Guangxuan Xu, Dinesh Raghu, Sachindra Joshi, Asim Munawar, and Ramón Fernandez Astudillo. Brain: Bayesian reward-conditioned amortized inference for natural language generation from feedback. In *ICML*. OpenReview.net, 2024.

[131] Jing-Cheng Pang, Pengyuan Wang, Kaiyuan Li, Xiong-Hui Chen, Jiacheng Xu, Zongzhang Zhang, and Yang Yu. Language model self-improvement by reinforcement learning contemplation. In *ICLR*. OpenReview.net, 2024.

[132] Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. Disentangling length from quality in direct preference optimization. In *ACL (Findings)*, pages 4998–5017. Association for Computational Linguistics, 2024.

[133] Jacob Parnell, Inigo Jauregi Unanue, and Massimo Piccardi. A multi-document coverage reward for relaxed multi-document summarization. In *ACL (1)*, pages 5112–5128. Association for Computational Linguistics, 2022.

[134] Miao Peng, Nuo Chen, Zongrui Suo, and Jia Li. Rewarding graph reasoning process makes llms more generalized reasoners. *CoRR*, abs/2503.00845, 2025.

[135] Moschoula Pternea, Prerna Singh, Abir Chakraborty, Yagna D. Oruganti, Mirco Milletarí, Sayli Bapat, and Kebei Jiang. The RL/LLM taxonomy tree: Reviewing synergies between reinforcement learning and large language models. *J. Artif. Intell. Res.*, 80:1525–1573, 2024.

[136] Han Qi, Haochen Yang, Qiaosheng Zhang, and Zhuoran Yang. Sample-efficient reinforcement learning from human feedback via information-directed sampling. *CoRR*, abs/2502.05434, 2025.

[137] Yuxiao Qu, Tianjun Zhang, Naman Garg, and Aviral Kumar. Recursive introspection: Teaching language model agents how to self-improve. In *NeurIPS*, 2024.

[138] Shanghaoran Quan. Dmoerm: Recipes of mixture-of-experts for effective reward modeling. In *ACL (Findings)*, pages 7006–7028. Association for Computational Linguistics, 2024.

[139] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

[140] Rajkumar Ramamurthy, Prithviraj Ammanabrolu, Kianté Brantley, Jack Hessel, Rafet Sifa, Christian Bauckhage, Hannaneh Hajishirzi, and Yejin Choi. Is reinforcement learning (not) for natural language processing: Benchmarks, baselines, and building blocks for natural language policy optimization. In *ICLR*. OpenReview.net, 2023.

[141] Alexandre Ramé, Nino Vieillard, Léonard Hussenot, Robert Dadashi, Geoffrey Cideron, Olivier Bachem, and Johan Ferret. WARM: on the benefits of weight averaged reward models. In *ICML*. OpenReview.net, 2024.

[142] Shyam Sundhar Ramesh, Yifan Hu, Iason Chaimalas, Viraj Mehta, Pier Giuseppe Sessa, Haitham Bou-Ammar, and Ilija Bogunovic. Group robust preference optimization in reward-free RLHF. In *NeurIPS*, 2024.

[143] Mathieu Rita, Florian Strub, Rahma Chaabouni, Paul Michel, Emmanuel Dupoux, and Olivier Pietquin. Countering reward over-optimization in LLM with demonstration-guided reinforcement learning. In *ACL (Findings)*, pages 12447–12472. Association for Computational Linguistics, 2024.

[144] Chris Samarinas and Hamed Zamani. Distillation and refinement of reasoning in small language models for document re-ranking. *CoRR*, abs/2504.03947, 2025.

[145] Dmitry Scherbakov, Nina C. Hubig, Vinita Jansari, Alexander Bakumenko, and Leslie A. Lenert. The emergence of large language models (LLM) as a tool in literature reviews: an LLM automated systematic review. *CoRR*, abs/2409.04600, 2024.

[146] Pier Giuseppe Sessa, Robert Dadashi-Tazehozi, Léonard Hussenot, Johan Ferret, Nino Vieillard, Alexandre Ramé, Bobak Shahriari, Sarah Perrin, Abram L. Friesen, Geoffrey Cideron, Sertan Girgin, Piotr Stanczyk, Andrea Michi, Danila Sinopalnikov, Sabela Ramos Garea, Amélie Héliou, Aliaksei Severyn, Matthew Hoffman, Nikola Momchev, and Olivier Bachem. BOND: aligning llms with best-of-n distillation. In *ICLR*. OpenReview.net, 2025.

[147] Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. Rewarding progress: Scaling automated process verifiers for LLM reasoning. In *ICLR*. OpenReview.net, 2025.

[148] Lior Shani, Aviv Rosenberg, Asaf B. Cassel, Oran Lang, Daniele Calandriello, Avital Zipori, Hila Noga, Orgad Keller, Bilal Piot, Idan Szpektor, Avinatan Hassidim, Yossi Matias, and Rémi Munos. Multi-turn reinforcement learning from preference human feedback. *CoRR*, abs/2405.14655, 2024.

[149] Ruichen Shao, Bei Li, Gangao Liu, Yang Chen, ZhouXiang, Jingang Wang, Xunliang Cai, and Peng Li. Earlier tokens contribute more: Learning direct preference optimization from temporal decay perspective. In *ICLR*. OpenReview.net, 2025.

[150] Wei Shen, Rui Zheng, WenYu Zhan, Jun Zhao, Shihan Dou, Tao Gui, Qi Zhang, and Xuanjing Huang. Loose lips sink ships: Mitigating length bias in reinforcement learning from human feedback. In *EMNLP (Findings)*, pages 2859–2873. Association for Computational Linguistics, 2023.

[151] Zhewen Shen, Aditya Joshi, and Ruey-Cheng Chen. BAMBINO-LM: (bilingual-)human-inspired continual pretraining of babylm. *CoRR*, abs/2406.11418, 2024.

[152] Wentao Shi, Mengqi Yuan, Junkang Wu, Qifan Wang, and Fuli Feng. Direct multi-turn preference optimization for language agents. In *EMNLP*, pages 2312–2324. Association for Computational Linguistics, 2024.

[153] Lei Shu, Liangchen Luo, Jayakumar Hoskere, Yun Zhu, Yinxiao Liu, Simon Tong, Jindong Chen, and Lei Meng. Rewritelm: An instruction-tuned large language model for text rewriting. In *AAAI*, pages 18970–18980. AAAI Press, 2024.

[154] Anand Siththaranjan, Cassidy Laidlaw, and Dylan Hadfield-Menell. Distributional preference learning: Understanding and accounting for hidden context in RLHF. In *ICLR*. OpenReview.net, 2024.

[155] Charlie Snell, Ilya Kostrikov, Yi Su, Sherry Yang, and Sergey Levine. Offline RL for natural language generation with implicit language Q learning. In *ICLR*. OpenReview.net, 2023.

[156] Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. Preference ranking optimization for human alignment. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan, editors, *AAAI*, pages 18990–18998. AAAI Press, 2024.

[157] Yuda Song, Gokul Swamy, Aarti Singh, J. Andrew Bagnell, and Wen Sun. The importance of online data: Understanding preference fine-tuning via coverage. In *NeurIPS*, 2024.

[158] Moritz Stephan, Alexander Khazatsky, Eric Mitchell, Annie S. Chen, Sheryl Hsu, Archit Sharma, and Chelsea Finn. RLVF: learning from verbal feedback without overgeneralization. In *ICML*. OpenReview.net, 2024.

[159] Zexu Sun, Yiju Guo, Yankai Lin, Xu Chen, Qi Qi, Xing Tang, Xiuqiang He, and Ji-Rong Wen. Uncertainty and influence aware reward model refinement for reinforcement learning from human feedback. In *ICLR*. OpenReview.net, 2025.

[160] Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented RLHF. In *ACL (Findings)*, pages 13088–13110.

Association for Computational Linguistics, 2024.

[161] Zhiqing Sun, Yikang Shen, Hongxin Zhang, Qinhong Zhou, Zhenfang Chen, David Daniel Cox, Yiming Yang, and Chuang Gan. SALMON: self-alignment with instructable reward models. In *ICLR*. OpenReview.net, 2024.

[162] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning - an introduction, 2nd Edition*. MIT Press, 2018.

[163] Wannita Takerngsaksiri, Rujikorn Charakorn, Chakkrit Tantithamthavorn, and Yuan-Fang Li. Pytester: Deep reinforcement learning for text-to-testcase generation. *J. Syst. Softw.*, 224:112381, 2025.

[164] Weihao Tan, Wentao Zhang, Shanqi Liu, Longtao Zheng, Xinrun Wang, and Bo An. True knowledge comes from practice: Aligning llms with embodied environments via reinforcement learning. *CoRR*, abs/2401.14151, 2024.

[165] Shuchang Tao, Liuyi Yao, Hanxing Ding, Yuexiang Xie, Qi Cao, Fei Sun, Jinyang Gao, Huawei Shen, and Bolin Ding. When to trust llms: Aligning confidence with response quality. In *ACL (Findings)*, pages 5984–5996. Association for Computational Linguistics, 2024.

[166] Elizaveta Tennant, Stephen Hailes, and Mirco Musolesi. Moral alignment for LLM agents. In *ICLR*. OpenReview.net, 2025.

[167] Luong Quoc Trung, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. Reft: Reasoning with reinforced fine-tuning. In *ACL (1)*, pages 7601–7614. Association for Computational Linguistics, 2024.

[168] Binghai Wang, Rui Zheng, Lu Chen, Zhiheng Xi, Wei Shen, Yuhao Zhou, Dong Yan, Tao Gui, Qi Zhang, and Xuanjing Huang. Reward modeling requires automatic adjustment based on data quality. In *EMNLP (Findings)*, pages 4041–4064. Association for Computational Linguistics, 2024.

[169] Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse KL: generalizing direct preference optimization with diverse divergence constraints. In *ICLR*. OpenReview.net, 2024.

[170] Chenglong Wang, Yang Gan, Yifu Huo, Yongyu Mu, Murun Yang, Qiaozhi He, Tong Xiao, Chunliang Zhang, Tongran Liu, and Jingbo Zhu. Rovrm: A robust visual reward model optimized via auxiliary textual preference data. In *AAAI*, pages 25336–25344. AAAI Press, 2025.

[171] Chenglong Wang, Hang Zhou, Yimin Hu, Yifu Huo, Bei Li, Tongran Liu, Tong Xiao, and Jingbo Zhu. ESRL: efficient sampling-based reinforcement learning for sequence generation. In *AAAI*, pages 19107–19115. AAAI Press, 2024.

[172] Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. Openchat: Advancing open-source language models with mixed-quality data. In *ICLR*. OpenReview.net, 2024.

[173] Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards. In *ACL (1)*, pages 8642–8655. Association for Computational Linguistics, 2024.

[174] Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *EMNLP (Findings)*, pages 10582–10592. Association for Computational Linguistics, 2024.

[175] Mingzhi Wang, Chengdong Ma, Qizhi Chen, Linjian Meng, Yang Han, Jiancong Xiao, Zhaowei Zhang, Jing Huo, Weijie J. Su, and Yaodong Yang. Magnetic preference optimization: Achieving last-iterate convergence for language model alignment. In *ICLR*. OpenReview.net, 2025.

[176] Ning Wang, Bingkun Yao, Jie Zhou, Yuchen Hu, Xi Wang, Zhe Jiang, and Nan Guan. Large language model for verilog generation with code-structure-guided reinforcement learning. In *2025 IEEE International Conference on LLM-Aided Design (ICLAD)*, pages 164–170. IEEE, 2025.

[177] Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *ACL (1)*, pages 9426–9439. Association for Computational Linguistics, 2024.

[178] Ruiyi Wang, Haofei Yu, Wenxin Zhang, Zhengyang Qi, Maarten Sap, Yonatan Bisk, Graham Neubig, and Hao Zhu. Sotopia-π: Interactive learning of socially intelligent language agents. In *ACL (1)*, pages 12912–12940. Association for Computational Linguistics, 2024.

[179] Shiqi Wang, Zhengze Zhang, Rui Zhao, Fei Tan, and Cam-Tu Nguyen. Reward difference optimization for sample reweighting in offline RLHF. In *EMNLP (Findings)*, pages 2109–2123. Association for Computational Linguistics, 2024.

[180] Shuhe Wang, Shengyu Zhang, Jie Zhang, Runyi Hu, Xiaoya Li, Tianwei Zhang, Jiwei Li, Fei Wu, Guoyin Wang, and Eduard H. Hovy. Reinforcement learning enhanced llms: A survey. *CoRR*, abs/2412.10400, 2024.

[181] Ziqi Wang, Le Hou, Tianjian Lu, Yuexin Wu, Yunxuan Li, Hongkun Yu, and Heng Ji. Enabling lanuguage models to implicitly learn self-improvement. In *ICLR*. OpenReview.net, 2024.

[182] Jiaxin Wen, Pei Ke, Hao Sun, Zhexin Zhang, Chengfei Li, Jinfeng Bai, and Minlie Huang. Unveiling the implicit toxicity in large language models. In *EMNLP*, pages 1322–1338. Association for Computational Linguistics, 2023.

[183] Muning Wen, Ziyu Wan, Jun Wang, Weinan Zhang, and Ying Wen. Reinforcing LLM agents via policy optimization with action decomposition. In *NeurIPS*, 2024.

[184] Yongjun Wen, Zhihao Cui, Yihao Liu, Zhao Zhang, Jiake Zhou, and Lijun Tang. UCP: a unified framework for code generation with pseudocode-based multi-task learning and reinforcement alignment. *J. Supercomput.*, 81(8):1010, 2025.

[185] Fan Wu, Huseyin A. Inan, Arturs Backurs, Varun Chandrasekaran, Janardhan Kulkarni, and Robert Sim. Privately aligning language models with reinforcement learning. In *ICLR*. OpenReview.net, 2024.

[186] Feijie Wu, Xiaoze Liu, Haoyu Wang, Xingchen Wang, Lu Su, and Jing Gao. Towards federated RLHF with aggregated client preference for llms. In *ICLR*. OpenReview.net, 2025.

[187] Jiaxing Wu, Lin Ning, Luyang Liu, Harrison Lee, Neo Wu, Chao Wang, Sushant Prakash, Shawn O'Banion, Bradley Green, and Jun Xie. RLPF: reinforcement learning from prediction feedback for user summarization with llms. In *AAAI*, pages 25488–25496. AAAI Press, 2025.

[188] Junda Wu, Xintong Li, Ruoyu Wang, Yu Xia, Yuxin Xiong, Jianing Wang, Tong Yu, Xiang Chen, Branislav Kveton, Lina Yao, Jingbo Shang, and Julian J. McAuley. OCEAN: offline chain-of-thought evaluation and alignment in large language models. In *ICLR*. OpenReview.net, 2025.

[189] Minghao Wu, Thuy-Trang Vu, Lizhen Qu, and Reza Haf. Mixture-of-skills: Learning to optimize data usage for fine-tuning large language models. In *EMNLP*, pages 14226–14240. Association for Computational Linguistics, 2024.

[190] Shujin Wu, Yi Fung, Sha Li, Yixin Wan, Kai-Wei Chang, and Heng Ji. MACAROON: training vision-language models to be your engaged partners. In *EMNLP (Findings)*, pages 7715–7731. Association for Computational Linguistics, 2024.

[191] Shujin Wu, Yi R. Fung, Cheng Qian, Jeonghwan Kim, Dilek Hakkani-Tur, and Heng Ji. Aligning llms with individual preferences via interaction. In *COLING*, pages 7648–7662. Association for Computational Linguistics, 2025.

[192] Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. In *ICLR*. OpenReview.net, 2025.

[193] Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for language model training. In *NeurIPS*, 2023.

[194] Markus Wulfmeier, Michael Bloesch, Nino Vieillard, Arun Ahuja, Jörg Bornschein, Sandy H. Huang, Artem Sokolov, Matt Barnes, Guillaume Desjardins, Alex Bewley, Sarah Bechtle, Jost Tobias Springenberg, Nikola Momchev, Olivier Bachem, Matthieu Geist, and Martin A. Riedmiller. Imitating language via scalable inverse reinforcement learning. In *NeurIPS*, 2024.

[195] Zhiheng Xi, Wenxiang Chen, Boyang Hong, Senjie Jin, Rui Zheng, Wei He, Yiwen Ding, Shichun Liu, Xin Guo, Junzhe Wang, Honglin Guo, Wei Shen, Xiaoran Fan, Yuhao Zhou, Shihan Dou, Xiao Wang, Xinbo Zhang, Peng Sun, Tao Gui, Qi Zhang, and Xuanjing Huang. Training large language models for reasoning through reverse curriculum reinforcement learning. In *ICML*. OpenReview.net, 2024.

[196] Han Xia, Songyang Gao, Qiming Ge, Zhiheng Xi, Qi Zhang, and Xuanjing Huang. Inverse-q*: Token level reinforcement learning for aligning large language models without preference data. In *EMNLP (Findings)*, pages 8178–8188. Association for Computational Linguistics, 2024.

[197] Yu Xia, Tong Yu, Zhankui He, Handong Zhao, Julian J. McAuley, and Shuai Li. Aligning as debiasing: Causality-aware alignment via reinforcement learning with interventional feedback. In *NAACL-HLT*, pages 4684–4695. Association for Computational Linguistics, 2024.

[198] Teng Xiao, Yige Yuan, Mingxiao Li, Zhengyu Chen, and Vasant G. Honavar. On a connection between imitation learning and RLHF. In *ICLR*. OpenReview.net, 2025.

[199] Shuo Xie, Fangzhi Zhu, Jiahui Wang, Lulu Wen, Wei Dai, Xiaowei Chen, Junxiong Zhu, Kai Zhou, and Bo Zheng. MPPO: multi pair-wise preference optimization for llms with arbitrary negative samples. In *COLING*, pages 1545–1554. Association for Computational Linguistics, 2025.

[200] Dehong Xu, Liang Qiu, Minseok Kim, Faisal Ladhak, and Jaeyoung Do. Aligning large language models via fine-grained supervision. In *ACL (Short Papers)*, pages 673–680. Association for Computational Linguistics, 2024.

[201] Tianyang Xu, Shujin Wu, Shizhe Diao, Xiaoze Liu, Xingyao Wang, Yangyi Chen, and Jing Gao. Sayself: Teaching llms to express confidence with self-reflective rationales. In *EMNLP*, pages 5985–5998. Association for Computational Linguistics, 2024.

[202] Diji Yang, Jinmeng Rao, Kezhen Chen, Xiaoyuan Guo, Yawen Zhang, Jie Yang, and Yi Zhang. IM-RAG: multi-round retrieval-augmented generation through learning inner monologues. In *SIGIR*, pages 730–740. ACM, 2024.

[203] Hongyu Yang, Jiahui Hou, Liyang He, and Rui Li. Multi-perspective preference alignment of llms for programming-community question answering. In *COLING*, pages 1667–1682. Association for Computational Linguistics, 2025.

[204] Jonghyeon Yang and Kiyun Yu. Reinforcement learning for constraint-aware knowledge base question answering. *IEEE Access*, 13:157294–157303, 2025.

[205] Kevin Yang, Dan Klein, Asli Celikyilmaz, Nanyun Peng, and Yuandong Tian. RLCD: reinforcement learning from contrastive distillation for LM alignment. In *ICLR*. OpenReview.net, 2024.

[206] Mengyuan Yang, Mengying Zhu, Yan Wang, Linxun Chen, Yilei Zhao, Xiuyuan Wang, Bing Han, Xiaolin Zheng, and Jianwei Yin. Fine-tuning large language model based explainable recommendation with explainable quality reward. In *AAAI*, pages 9250–9259. AAAI Press, 2024.

[207] Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. Regularizing hidden states enables learning generalizable reward model for llms. In *NeurIPS*, 2024.

[208] Shuo Yang and Gjergji Kasneci. Is crowdsourcing breaking your bank? cost-effective fine-tuning of pre-trained language models with proximal policy optimization. In *LREC/COLING*, pages 9304–9314. ELRA and ICCL, 2024.

[209] Hai Ye and Hwee Tou Ng. Preference-guided reflective sampling for aligning language models. In *EMNLP*, pages 21646–21668. Association for Computational Linguistics, 2024.

[210] Yaowen Ye, Cassidy Laidlaw, and Jacob Steinhardt. Iterative label refinement matters more than preference optimization under weak supervision. In *ICLR*. OpenReview.net, 2025.

[211] Eunseop Yoon, Hee Suk Yoon, SooHwan Eom, Gunsoo Han, Daniel Wontae Nam, Daejin Jo, Kyoung-Woon On, Mark Hasegawa-Johnson, Sungwoong Kim, and Chang Dong Yoo. TLCR: token-level continuous reward for fine-grained reinforcement learning from human feedback. In *ACL (Findings)*, pages 14969–14981. Association for Computational Linguistics, 2024.

[212] Xiao Yu, Qingyang Wu, Kun Qian, and Zhou Yu. KRLS: improving end-to-end response generation in task oriented dialog with reinforced keywords learning. In *EMNLP*, pages 12338–12358. Association for Computational Linguistics, 2023.

[213] Xiaohan Yu, Li Zhang, and Chong Chen. Explainable CTR prediction via LLM reasoning. In *WSDM*, pages 707–716. ACM, 2025.

[214] Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. RRHF: rank responses to align language models with human feedback. In *NeurIPS*, 2023.

[215] Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. Rest-mcts*: LLM self-training via process reward guided tree search. In *NeurIPS*, 2024.

[216] Han Zhang, Yu Lei, Lin Gui, Min Yang, Yulan He, Hui Wang, and Ruifeng Xu. CPPO: continual learning for reinforcement learning with human feedback. In *ICLR*. OpenReview.net, 2024.

[217] Hanning Zhang, Pengcheng Wang, Shizhe Diao, Yong Lin, Rui Pan, Hanze Dong, Dylan Zhang, Pavlo Molchanov, and Tong Zhang. Entropy-regularized process reward model. *Trans. Mach. Learn. Res.*, 2025, 2025.

[218] Hongbo Zhang, Junying Chen, Feng Jiang, Fei Yu, Zhihong Chen, Guiming Chen, Jianquan Li, Xiangbo Wu, Zhiyi Zhang, Qingying Xiao, Xiang Wan, Benyou Wang, and Haizhou Li. Huatuogpt, towards taming language model to be a doctor. In *EMNLP (Findings)*, pages 10859–10885. Association for Computational Linguistics, 2023.

[219] Jinghan Zhang, Xiting Wang, Yiqiao Jin, Changyu Chen, Xinhao Zhang, and Kunpeng Liu. Prototypical reward network for data-efficient RLHF. In *ACL (1)*, pages 13871–13884. Association for Computational Linguistics, 2024.

[220] Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Pengfei Li, Yu Fu, Xingtai Lv, Yuchen Zhang, Sihang Zeng, Shang Qu, Haozhan Li, Shijie Wang, Yuru Wang, Xinwei Long, Fangfu Liu, Xiang Xu, Jiaze Ma, Xuekai Zhu, Ermo Hua, Yihao Liu, Zonglin Li, Huayu Chen, Xiaoye Qu, Yafu Li, Weize Chen, Zhenzhao Yuan, Junqi Gao, Dong Li, Zhiyuan Ma, Ganqu Cui, Zhiyuan Liu, Biqing Qi, Ning Ding, and Bowen Zhou. A survey of reinforcement learning for large reasoning models. *CoRR*, abs/2509.08827, 2025.

[221] Mengxi Zhang, Wenhao Wu, Yu Lu, Yuxin Song, Kang Rong, Huanjin Yao, Jianbo Zhao, Fanglong Liu, Haocheng Feng, Jingdong Wang, and Yifan Sun. Automated multi-level preference for mllms. In *NeurIPS*, 2024.

[222] Qining Zhang and Lei Ying. Zeroth-order policy gradient for reinforcement learning from human feedback without reward inference. In *ICLR*. OpenReview.net, 2025.

[223] Shaowei Zhang and Deyi Xiong. Backmath: Towards backward reasoning for solving math problems step by step. In *COLING (Industry)*, pages 466–482. Association for Computational Linguistics, 2025.

[224] Shenao Zhang, Donghan Yu, Hiteshi Sharma, Han Zhong, Zhihan Liu, Ziyi Yang, Shuohang Wang, Hany Hassan Awadalla, and Zhaoran Wang. Self-exploring language models: Active preference elicitation for online alignment. *Trans. Mach. Learn. Res.*, 2025, 2025.

[225] Tianjun Zhang, Fangchen Liu, Justin Wong, Pieter Abbeel, and Joseph E. Gonzalez. The wisdom of hindsight makes language models better instruction followers. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 41414–41428. PMLR, 2023.

[226] Wenxuan Zhang, Philip Torr, Mohamed Elhoseiny, and Adel Bibi. Bi-factorial preference optimization: Balancing safety-helpfulness in language models. In *ICLR*. OpenReview.net, 2025.

[227] Xiaoying Zhang, Jean-Francois Ton, Wei Shen, Hongning Wang, and Yang Liu. Mitigating reward overoptimization via lightweight uncertainty estimation. In *NeurIPS*, 2024.

[228] Yuheng Zhang, Dian Yu, Baolin Peng, Linfeng Song, Ye Tian, Mingyue Huo, Nan Jiang, Haitao Mi, and Dong Yu. Iterative nash policy optimization: Aligning llms with general preferences via no-regret learning. In *ICLR*. OpenReview.net, 2025.

[229] Zongmeng Zhang, Yufeng Shi, Jinhua Zhu, Wengang Zhou, Xiang Qi, Peng Zhang, and Houqiang Li. Trustworthy alignment of retrieval-augmented large language models via reinforcement learning. In *ICML*. OpenReview.net, 2024.

[230] Rui Zheng, Wei Shen, Yuan Hua, Wenbin Lai, Shihan Dou, Yuhao Zhou, Zhiheng Xi, Xiao Wang, Haoran Huang, Tao Gui, Qi Zhang, and Xuanjing Huang. Improving generalization of alignment with human preferences through group invariant learning. In *ICLR*. OpenReview.net, 2024.

[231] Artem Zholus, Maksim Kuznetsov, Roman Schutski, Rim Shayakhmetov, Daniil Polykovskiy, Sarath Chandar, and Alex Zhavoronkov. Bindgpt: A scalable framework for 3d molecular design via language modeling and reinforcement learning. In *AAAI*, pages 26083–26091. AAAI Press, 2025.

[232] Hang Zhou, Chenglong Wang, Yimin Hu, Tong Xiao, Chunliang Zhang, and Jingbo Zhu. Prior constraints-based reward model training for aligning large language models. In *CCL*, volume 14761 of *Lecture Notes in Computer Science*, pages 555–570. Springer, 2024.

[233] Jiayi Zhou, Jiaming Ji, Josef Dai, and Yaodong Yang. Sequence to sequence reward modeling: Improving RLHF by language feedback. In *AAAI*, pages 27765–27773. AAAI Press, 2025.

[234] Runlong Zhou, Simon S. Du, and Beibin Li. Reflect-rl: Two-player online RL fine-tuning for lms. In *ACL (1)*, pages 995–1015. Association for Computational Linguistics, 2024.

[235] Wenxuan Zhou, Ravi Agrawal, Shujian Zhang, Sathish Reddy Indurthi, Sanqiang Zhao, Kaiqiang Song, Silei Xu, and Chenguang Zhu. WPO: enhancing RLHF with weighted preference optimization. In *EMNLP*, pages 8328–8340. Association for Computational Linguistics, 2024.

[236] Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. Archer: Training language model agents via hierarchical multi-turn RL. In *ICML*. OpenReview.net, 2024.

[237] Zhanhui Zhou, Jie Liu, Jing Shao, Xiangyu Yue, Chao Yang, Wanli Ouyang, and Yu Qiao. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. In *ACL (Findings)*, pages 10586–10613. Association for Computational Linguistics, 2024.

[238] Banghua Zhu, Michael I. Jordan, and Jiantao Jiao. Iterative data smoothing: Mitigating reward overfitting and overoptimization in RLHF. In *ICML*. OpenReview.net, 2024.

[239] Liang Zhu, Feiteng Fang, Yuelin Bai, Longze Chen, Zhexiang Zhang, Minghuan Tan, and Min Yang. DEFT: distribution-guided efficient fine-tuning for human alignment. In *EMNLP (Findings)*, pages 15318–15331. Association for Computational Linguistics, 2024.

[240] Mingye Zhu, Yi Liu, Lei Zhang, Junbo Guo, and Zhendong Mao. LIRE: listwise reward enhancement for preference alignment. In *ACL (Findings)*, pages 3377–3394. Association for Computational Linguistics, 2024.