UNIVERSIDADE FEDERAL DE GOIÁS / INSTITUTO DE INFORMÁTICA

Aprendizado por Reforço para Alinhamento de LLMs

Estudo Comparativo de Estratégias de Recompensa Pós-Destilação

Lisandra Cristina de Moura Menezes

UFG
UNIVERSIDADE
FEDERAL DE GOIÁS

UNIVERSIDADE FEDERAL DE GOIÁS (UFG)
INSTITUTO DE INFORMÁTICA (INF)

LISANDRA CRISTINA DE MOURA MENEZES

Aprendizado por Reforço para Alinhamento de LLMs

Estudo Comparativo de Estratégias de Recompensa Pós-Destilação

Goiânia
2025

UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TRABALHO DE CONCLUSÃO DE CURSO DE GRADUAÇÃO NO REPOSITÓRIO INSTITUCIONAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio do Repositório Institucional (RI/UFG), regulamentado pela Resolução CEPEC no 1240/2014, sem ressarcimento dos direitos autorais, de acordo com a Lei no 9.610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo dos Trabalhos de Conclusão dos Cursos de Graduação disponibilizado no RI/UFG é de responsabilidade exclusiva dos autores. Ao encaminhar(em) o produto final, o(s) autor(a)(es)(as) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do Trabalho de Conclusão de Curso de Graduação (TCCG)

Nome(s) completo(s) do(a)(s) autor(a)(es)(as): LISANDRA CRISTINA DE MOURA MENEZES

Título do trabalho: Aprendizado por Reforço para Alinhamento de LLMs

Estudo Comparativo de Estratégias de Recompensa Pós-Destilação

2. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador) Concorda com a liberação total do documento [X] SIM [ ] NÃO[1]

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante: a) consulta ao(à)(s) autor(a)(es)(as) e ao(à) orientador(a); b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo do TCCG. O documento não será disponibilizado durante o período de embargo.

Casos de embargo:

- Solicitação de registro de patente;

- Submissão de artigo em revista científica;

- Publicação como capítulo de livro.

Obs.: Este termo deve ser assinado no SEI pelo orientador e pelo autor.

Documento assinado eletronicamente por **Lisandra Cristina De Moura Menezes**, **Discente**, em 04/02/2026, às 16:11, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

Documento assinado eletronicamente por **Fernando Marques Federson**, **Professor do Magistério Superior**, em 13/03/2026, às 11:36, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5956596** e o código CRC **C7497E71**.

Referência: Processo nº 23070.005511/2026-37 SEI nº 5956596

LISANDRA CRISTINA DE MOURA MENEZES

Aprendizado por Reforço para Alinhamento de LLMs

Estudo Comparativo de Estratégias de Recompensa Pós-Destilação

Relatório final de Trabalho de Conclusão de Curso, apresentado à Universidade Federal de Goiás, como parte das exigências para a obtenção do título de Bacharel em Inteligência Artificial.

Orientador: Prof. Dr. Fernando Marques Federson

Goiânia

2025

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UFG.

MENEZES, LISANDRA CRISTINA DE MOURA
Aprendizado por Reforço para Alinhamento de LLMs [manuscrito]: Estudo Comparativo de Estratégias de Recompensa Pós-Destilação / LISANDRA CRISTINA DE MOURA MENEZES. - 2025.
91 f.: 2025

Orientador: Prof. Dr. Fernando Marques Federson
Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Goiás, Instituto de Informática (INF), Inteligência Artificial, Goiânia, 2025.

1. Inteligência Artificial. 2. Aprendizado por Reforço. 3. Modelo de Linguagem Grande.

I. Federson, Fernando Marques , orient. II. Título.

CDU 004

LISANDRA CRISTINA DE MOURA MENEZES

Aprendizado por Reforço para Alinhamento de LLMs
Estudo Comparativo de Estratégias de Recompensa Pós-Destilação

Relatório final de Trabalho de Conclusão de Curso, apresentado à Universidade Federal de Goiás, como parte das exigências para a obtenção do título de Bacharel em Inteligência Artificial.

Data da Aprovação: 09 de dezembro de 2025.

Prof. Dr. Fernando Marques Federson
Orientador (INF-UFG)

Prof. Dr. Aldo André Díaz Salazar
Coordenador de TCC do BIA (INF-UFG)

Prof. Dr. Anderson da Silva Soares
Coordenador do BIA (INF-UFG)

Prof. Dr. Sávio Salvarino Teles de Oliveira
(INF-UFG)

LISANDRA CRISTINA DE MOURA MENEZES

Aprendizado por Reforço para Alinhamento de LLMs

Estudo Comparativo de Estratégias de Recompensa Pós-Destilação

RESUMO

Este Relatório de Conclusão de Curso tem como objetivo reunir os resultados da minha jornada para me tornar um especialista em **Alinhamento e Otimização de LLMs via Aprendizado por Reforço**. Uma ilustração e sua narrativa descrevem os períodos de trabalho. Os Apêndices contêm os Termos de Aceite de Entrega e os resultados obtidos durante cada período de trabalho.

Palavras-chave: Inteligência artificial; Aprendizado por reforço; Modelo de linguagem grande.

ABSTRACT

This Course Completion Report aims to bring together the results of my journey to become an expert in **Alignment and Optimization of LLMs via Reinforcement Learning**. An illustration and its narrative describe the work periods. The Appendices contain the Delivery Acceptance Terms and the results obtained during each work period.

Keywords: Artificial intelligence; Reinforcement learning; Large language model.

Goiânia

2025

Minha Jornada

Lisandra Cristina de Moura Menezes

Especialista em: Alinhamento e Otimização de LLMs via Aprendizado por Reforço

MINHA JORNADA

Nome: **Lisandra Cristina de Moura Menezes**

Especialidade: **Alinhamento e Otimização de LLMs via Aprendizado por Reforço**

Objetivo deste documento

Durante o processo da disciplina Residência em IA[1], foram gerados diversos resultados na construção da minha especialização. A cada semana, um conjunto de resultados foi formalizado por um Termo de Aceite de Entrega e avaliado por uma banca, considerando o planejado e o realizado para o período. Este documento tem como objetivo descrever esses resultados obtidos, fazendo referência aos Termos de Aceite de Entrega e seus documentos associados.

Minha Jornada

Minha jornada se iniciou com atividade para definição da área de conhecimento da minha especialização. Durante a Residência, foi sugerido explorar a estrutura temática da *27th International Conference on Artificial Intelligence* (ICAI'25), o que me ajudou a compreender o posicionamento do **Aprendizado por Reforço** ou *Reinforcement Learning* (RL) dentro do campo mais amplo da Inteligência Artificial. No entanto, foi ao analisar os trabalhos aceitos na *International Conference on Learning Representations* (ICLR 2025) que percebi onde estava a fronteira de pesquisa que me interessava: a aplicação emergente de RL em **Modelos de Linguagem de Grande Escala** (LLMs). Além disso, outras experiências foram importantes para mim, por exemplo, a disciplina de Aprendizado de Máquina por Reforço, minha experiência e curiosidade em Teoria do Conhecimento, Psicologia e Filosofia

[1] Dez Semanas, entre setembro de 2025 e dezembro de 2025.

da Mente e uma exposição inicial a projetos do CEIA[2] também relacionados a RL e cativaram meu interesse pela área.

Nas **Semanas 1 a 3**, as leituras de artigos e blogs técnicos consolidaram minha decisão de investigar a intersecção entre RL e LLMs, especialmente nas abordagens *Reinforcement Learning with Human Feedback* (RLHF) e *Reinforcement Learning with AI Feedback* (RLAIF). O trabalho de HUANG et al.[3] foi particularmente relevante, ao demonstrar que o RL pode tanto mitigar quanto intensificar alucinações, reforçando a necessidade de uma compreensão crítica para evitar aplicações inadequadas de etapas adicionais de alinhamento. A partir dessas leituras, tornou-se igualmente clara a distinção entre RLHF baseado em anotadores humanos e RLAIF, que utiliza modelos para fornecer feedback em maior escala. Por fim, reúno no **Apêndice 1** as explorações iniciais, as tabelas de artigos e os fichamentos referentes às leituras realizadas neste período.

A **Semana 4** representou um ponto de virada: deixei de analisar métodos como *Supervised Fine-Tuning* (SFT), *Direct Preference Optimization* (DPO), *Proximal Policy Optimization* (PPO) e *Group Relative Policy Optimization* (GRPO) de forma isolada e passei a compreender a área a partir de três componentes fundamentais: (1) a fonte de dados, (2) a função de recompensa e (3) o algoritmo responsável por transformar esses elementos em coeficientes de gradiente. Os trabalhos de Wang et al.[4] e Shao et al.[5] introduziram essa forma estruturada de visualizar o fluxo geral do **alinhamento do LLM via RL**, e, a partir do framework estabelecido por Ouyang et al.[6], desenvolvi uma adaptação que também incorporava abordagens de RLAIF. Com essa reorganização conceitual, pude elaborar uma tabela comparativa e um framework de aplicação prática para RL em LLMs, ambos apresentados no **Apêndice 2**. Outro aspecto particularmente interessante emergiu durante

[2] Centro de Excelência em Inteligência Artificial (UFG).
[3] HUANG, L. et al. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. arXiv preprint arXiv:2311.05232, 2023. Disponível em: https://arxiv.org/pdf/2311.05232.
[4] WANG, Shuhe et al. *Reinforcement Learning Enhanced LLMs: A Survey*. arXiv, v-preprint, 2024. Disponível em: https://arxiv.org/abs/2412.10400.
[5] DEEPSEEK-AI et al. *DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models*. arXiv, v-preprint, 2024. Disponível em: https://arxiv.org/abs/2402.03300.
[6] OUYANG, Long et al. *Training Language Models to Follow Instructions with Human Feedback*. arXiv, v-preprint, 2022. Disponível em: https://arxiv.org/abs/2203.02155.

essa **Semana**. Compreendi que o RL não adicionava conhecimento novo ao modelo, mas apenas reorganizava sua distribuição de probabilidades. Essa constatação redefiniu minha compreensão do processo e levou-me a preferir os termos **melhoramento**, **alinhamento** ou **otimização**, por representarem de forma mais precisa o papel do RL nesse contexto.

Nas **Semanas 5 e 6**, mergulhei no estudo das diferentes estratégias de RLAIF identificadas na literatura. O survey organizado por Wang et al.[7] apresentou três abordagens principais que explorei sistematicamente. A primeira, Destilação de Datasets, envolvia usar um LLM avançado como GPT-4 para anotar milhares de exemplos, gerando preferências artificiais que depois alimentavam o treinamento de um modelo de recompensa. Explorei o dataset UltraFeedback como exemplo prático, onde cada instrução tinha múltiplas respostas geradas por diferentes modelos, todas avaliadas por GPT-4 em dimensões como *helpfulness*, *honesty* e *instruction-following*. A segunda abordagem, *Prompt Rewarding*, eliminava a necessidade de treinar um modelo de recompensa separado. O LLM atuava como função de recompensa diretamente via API, avaliando respostas através de prompts estruturados - o chamado *LLM-as-a-Judge*. A terceira abordagem, *Self-rewarding*, propunha que o próprio modelo em treinamento gerasse e avaliasse suas respostas, criando um ciclo de auto-aperfeiçoamento. Cabe ressaltar que o artigo Yuan et al.[8] argumenta que isso poderia levar os LLMs a capacidades sobre-humanas, já que cada iteração teria acesso a feedbacks de melhor qualidade que a anterior. Evidenciou-se que cada abordagem apresentava trade-offs relevantes. A destilação proporciona escalabilidade e consistência, porém requeria um pipeline complexo e oneroso, envolvendo geração de dados, treinamento de um modelo de recompensa e posterior aplicação no processo de RL. A técnica de *prompt rewarding* oferece elevada flexibilidade e facilidade de ajuste de critérios, mas dependia de chamadas de API, o que implicava custos adicionais e possíveis limitações de latência. Por sua vez, o *self-rewarding* propunha maior autonomia ao modelo, embora apresentasse risco significativo de mode collapse[9], situação em que o sistema passava a avaliar positivamente suas próprias respostas mesmo quando estas não apresentavam qualidade objetiva.

[7] Idem 3
[8] YUAN, Weizhe et al. *Self-Rewarding Language Models*. arXiv, v-preprint, 2024. Disponível em: https://arxiv.org/abs/2401.10020.
[9] Fenômeno onde o modelo converge para gerar respostas limitadas e repetitivas, perdendo diversidade.

Durante essa etapa, também avancei na construção das tabelas comparativas e na organização de um repositório contendo as funções de recompensa e as estratégias analisadas. O **Apêndice 3** reúne os resultados obtidos neste período.

Nas **Semanas 7 e 8**, concentrei-me em converter o conhecimento acumulado em um plano experimental. Minha intenção inicial era reproduzir o método apresentado por Yuan et al.[10], devido ao seu potencial de autonomia e melhoria iterativa. Entretanto, após analisar repositórios produzidos pela comunidade e detalhar os requisitos técnicos, concluí que a reprodução não seria possível neste momento. O trabalho original utilizava o modelo Llama 2 70B e requeria múltiplas iterações envolvendo geração massiva de prompts, avaliação via *LLM-as-a-Judge*, construção de pares de preferência e re-treinamento com DPO, cada ciclo demandando vários dias de GPU, além de depender de filtros de dados e hiperparâmetros não disponibilizados no artigo. Diante da inviabilidade do experimento anterior, optei por direcionar o estudo para os modelos destilados do DeepSeek-R1 apresentados pela DeepSeek-AI[11]. No artigo, foram apresentadas três variantes: (1) DeepSeek-R1-Zero; (2) DeepSeek-R1; (3) e os modelos destilados - nos quais os pesquisadores criaram um conjunto de dados a partir das respostas dos modelos maiores e destilaram esse conhecimento em modelos menores por meio de *Supervised Fine-Tuning* (SFT). Os autores relataram que o uso isolado de RL em modelos menores não apresenta ganhos significativos, enquanto a etapa de destilação conduz a resultados mais consistentes. Eles também deixam em aberto a investigação sobre a aplicação adicional de RL após a etapa de SFT. Essa lacuna configurava uma oportunidade pertinente para avaliar se o RL pós-destilação poderia aprimorar capacidades de raciocínio. Os modelos destilados estão disponíveis publicamente em tamanhos acessíveis (7B–8B) e pesquisadores da Hugging Face disponibilizam o framework Open-R1, que reproduz diretamente o *pipeline* do DeepSeek-R1. Com base nisso, estabeleci a avaliação para o experimento em que utilizaria

[10] Idem 8

[11] DEEPSEEK-AI et al. *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*. arXiv, v-preprint, 2025. Disponível em: https://arxiv.org/abs/2501.12948.

as 15 primeiras questões do AIME 2025[12] e a métrica *pass@k*[13] (k = 1 e k = 8). Conforme registrado no **Apêndice 4**, avaliei os modelos Llama-8B e Qwen-7B, obtendo os seguintes resultados: Llama Pass@1 de 20% e Pass@8 de 60%; Qwen Pass@1 de 26,67% e Pass@8 de 40%. Optei por prosseguir com o Llama-8B devido ao seu desempenho superior em Pass@8, o que indicava uma maior potencial latente de melhoria.

A **Semana 9** foi dedicada ao primeiro experimento, cujo objetivo era aplicar RL para aprimorar o raciocínio lógico-matemático. Para isso, utilizei cinco funções de recompensa: (1) *accuracy_reasoning*, que verificava se a resposta final estava correta; (2) *reasoning_steps_reward*, que avaliava a presença de raciocínio passo a passo; (3) *tag_count_reward*, que contabilizava o uso das tags *think* e *answer*; (4) *format_reward*, que verificava o uso correto dessas tags; e (5) *len_reward*, que penalizava respostas excessivamente longas para evitar o "pseudo-raciocínio". O experimento foi realizado com o dataset ReClor, voltado à lógica e compreensão textual, com treinamento de 250 *steps* utilizando 1000 amostras. Acompanhei o treinamento visualizando os gráficos no *Weights & Biases* e identifiquei alguns problemas significativos: a *len_reward* apresentou forte instabilidade, oscilando entre valores positivos e negativos; a *reasoning_steps_reward* também se mostrou "errática"; e a *format_reward* permaneceu zerada ao longo de todo o processo. Apenas a *tag_count_reward* e a *accuracy_reasoning* apresentaram alguma convergência. Os resultados finais confirmaram essas preocupações. O Pass@1 aumentou apenas de 20% para 26,67%, enquanto o Pass@8 piorou de 60% para 46,67%. Esses resultados sugerem que o modelo aprendera heurísticas específicas do conjunto de treinamento, sem generalização adequada. Com a observação do treinamento pude confirmar um ponto recorrente na literatura: recompensas mal definidas prejudicavam o processo de RL, e, neste caso, recompensas negativas mais fortes que as positivas provavelmente levaram o modelo a explorar excessivamente, desviando-se da política original do LLM. O **Apêndice 5** apresenta o experimento realizado e seus resultados.

[12] *American Invitational Mathematics Examination* de 2025 usado como *benchmark* para avaliação de raciocínio.

[13] Métrica que avalia a probabilidade de um modelo gerar pelo menos uma resposta correta em k tentativas.

Na **Semana 10**, conduzi o segundo experimento com o objetivo de aplicar RL para alinhar o modelo a preferências de qualidade adotando uma estratégia observada durante a minha jornada. Acrescentei ao treinamento, além das recompensas por raciocínio, tirando as que não estavam bem estruturadas, uma abordagem holística utilizando o *LLM-as-a-Judge* via API para avaliar correção, clareza, qualidade do raciocínio e aderência às instruções. Mantive o dataset ReClor para garantir comparabilidade direta com o experimento da **Semana 9**. A revisão do artigo "*RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback"* forneceu evidências de que LLMs avançados, como o GPT-4, poderiam alcançar aproximadamente 80% de concordância com preferências humanas, um nível semelhante ao observado entre avaliadores humanos. O comportamento do treinamento diferiu substancialmente do observado no primeiro experimento: a métrica *Kullback-Leibler divergence* (*KL-divergence*)[14] permaneceu baixa e estável, indicando que o modelo não se afastou indevidamente da distribuição original. Os resultados obtidos pela aplicação no AIME 2025 foram expressivos. O Pass@1 aumentou de 20% para 40%, representando um ganho de 20 pontos percentuais, três vezes superior ao observado no primeiro experimento. O Pass@8, embora tenha diminuído de 60% para 53,33%, apresentou uma queda significativamente menor do que a verificada na **Semana 9**, indicando maior estabilidade e generalização da política aprendida. O **Apêndice 6** apresenta este segundo experimento e seus resultados.

[14] Métrica que mede o quanto a distribuição de probabilidades do modelo se afastou da distribuição original. Valores baixos indicam que o modelo mantém comportamento próximo ao pré-treinamento; valores altos sinalizam mudanças drásticas que podem comprometer o conhecimento prévio.

APÊNDICE 1

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 3 de set. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

1. **Mapeamento da área em conferências internacionais**

No estudo sobre a estrutura temática do ICAI'25 - The 27th Int'l Conf on Artificial Intelligence identifiquei o objeto da minha pesquisa e montei uma classificação na qual o objeto se encontra.

- Nível 1 – Inteligência Artificial (IA)
- Nível 2 – Aprendizado de Máquina (Machine Learning)
- Nível 3 – Técnicas específicas - Reinforcement Learning (RL)
- Nível 4 – Aplicações de RL
    - 4.1 RL em Large Language Models (LLMs)
        - 4.1.1 RLHF (Reinforcement Learning with Human Feedback)
        - 4.1.2 RLAIF (Reinforcement Learning with AI Feedback)
        - 4.1.3 Algoritmos como PPO e variantes mais recentes (ex.: GRPO) para ajuste fino de modelos de linguagem

2. **Integração em comunidades e grupos de estudo**

Contato com pares (residência, mestrado e doutorado) da área de RL, buscando trocar experiências, estratégias de leitura e perspectivas de pesquisa.

3. **Levantamento de referências canônicas**

Identificação e leitura inicial de obras clássicas e autores consolidados, como Richard Sutton e Andrew Barto, para formar uma base teórica sólida sobre Aprendizado por Reforço.

4. **Busca por revisões e materiais de apoio**

Pesquisa de artigos de revisão, blogs técnicos e tutoriais que explicam terminologias usuais, funcionamento de algoritmos e a evolução recente do RL em LLMs.

5. **Definição de tópicos e algoritmos de interesse**

Levantamento de algoritmos centrais a serem estudados, incluindo: Q-Learning, SARSA, Actor-Critic, Policy Gradient, PPO, **GRPO**, além das abordagens modernas de RLHF e RLAIF.

DOCUMENTAÇÃO COMPLETA

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Aprofundar conhecimento em RL e produzir fichamento sobre os seguintes artigos, livros e blogs:

1. Reinforcement Learning: An Introduction de Richard Sutton e Andrew Barto - base teórica clássica sobre RL.
2. Deep Reinforcement Learning: An Overview - panorama da evolução do RL e contexto atual.
3. A Brief Survey of Deep Reinforcement Learning - resumo conciso com foco em aplicações modernas.

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Especialização

A pesquisa em “Aprendizado por Reforço aplicado a Modelos de Linguagem de Grande Escala (LLMs)” emerge como um tema relevante, especialmente a partir de trabalhos recentes como o DeepSeek, que demonstrou a possibilidade de emergência de raciocínio matemático por meio de técnicas de reforço, e de conferências, como a *International Conference on Learning Representations* (ICLR 2025), em que é possível observar artigos sobre o tema sendo publicados.

Como primeira atividade desenvolvida na residência, deveríamos entender o posicionamento da nossa área de interesse no campo da Ciência da Computação, sobretudo, para compreendermos os conhecimentos anteriores que culminaram nessa especialização e termos abrangência de conhecimento sobre o campo de interesse. Mapeei, inicialmente, o campo da seguinte maneira:

1. **Nível 1 – Inteligência Artificial (IA)**
2. **Nível 2 – Aprendizado de Máquina (Machine Learning)**
3. **Nível 3 – Técnicas específicas - Reinforcement Learning (RL)**
4. **Nível 4 – Aplicações de RL**
    a. **RL em Large Language Models (LLMs)**
        i. RLHF (Reinforcement Learning with Human Feedback)
        ii. RLAIF (Reinforcement Learning with AI Feedback)
        iii. Algoritmos como PPO e variantes mais recentes (ex.: GRPO) para ajuste fino de modelos de linguagem

Baseei-me na ICAI'25 - The 27th Int'l Conf on Artificial Intelligence e em conversas com outros especialistas para hierarquizar os conhecimentos em uma representação que possibilitasse enxergar as bases e as ramificações do conhecimento que gostaria de adquirir. Portanto, o **objeto de estudo** da especialização é a interseção de **RL em LLMs**, com ênfase nos algoritmos variantes do PPO e no uso de feedback humano e de LLMs

secundários como estratégia para aprimorar o aprendizado e a capacidade de raciocínio dos modelos.

Breve História

Richard Sutton e Andrew Barto são considerados os criadores do aprendizado por reforço moderno, pois propuseram que um agente interagindo com um ambiente desconhecido, onde ele não conhece nem as transições nem as recompensas, pode aprender pela exploração do ambiente, experimentando ações e ajustando suas políticas com base no feedback recebido.

Essa forma de aprendizado de máquina tem associação direta com a psicologia do comportamento, ou behaviorismo. O behaviorismo é uma abordagem sistemática no campo da psicologia que buscou compreender a cognição humana a partir do comportamento e da reação a estímulos do ambiente. B.F. Skinner, um dos principais teóricos behavioristas, argumentou que nossas ações são moldadas pelas consequências do que experimentamos. Segundo essa perspectiva, se conhecemos as forças externas, isto é, os mecanismos de recompensa e punição, podemos prever e até moldar comportamentos. Através do condicionamento, respostas que inicialmente eram voluntárias tornam-se involuntárias.

A similaridade de conceitos entre behaviorismo e aprendizado por reforço não é acidental. Sutton graduou-se em Psicologia e, junto com Barto, estudava o funcionamento dos neurônios e mecanismos de aprendizado animal na Universidade de Massachusetts. A partir dessa base interdisciplinar, eles formalizaram o aprendizado por reforço como um campo da Ciência da Computação. Com essa formalização, o campo desenvolveu frameworks teóricos e algoritmos computacionais que operacionalizam conceitos como agente, ambiente, política, recompensa e função de valor, estabelecendo mecanismos sistemáticos para treinar modelos através dessa abordagem de aprendizado por tentativa e erro.

No boom dos modelos de linguagem, o aprendizado por reforço ganhou o papel de alinhar as probabilidades de uma política às preferências humanas; um caso famoso disso

foi o GPT-3.5 Instruct. Contudo, em 2025, o artigo *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*[15] ganhou destaque por aplicar aprendizado por reforço especificamente para aprimoramento de capacidades de raciocínio. Em especial, durante o desenvolvimento do DeepSeek-R1-Zero, um modelo treinado exclusivamente com RL, o grupo conseguiu fazer emergir habilidades de raciocínio matemático com eficiência computacional e custos relativamente baixos.

Outro avanço técnico importante da organização DeepSeek-AI foi o desenvolvimento do Group Relative Policy Optimization (GRPO), uma variante do PPO que normaliza recompensas dentro de um grupo de saídas para economizar recursos computacionais no ajuste de modelos de linguagem por RL, conforme apresentado no artigo *DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models*[16].

Diante desse panorama relevante, a especialização teve como objetivo estudar artigos e modelos sobre aplicação de RL em LLMs, bem como compreender a base teórica do RL.

[15] Idem 11
[16] Idem 5

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 10 de set. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

1. Contexto e motivação

- Avanços recentes em RL: modelos da família R1 e artigos recentes mostram ganhos expressivos em generalização e raciocínio, consolidando a área como um campo emergente e relevante.
- ToolRL (Reward is All Tool Learning Needs): destaca a importância do desenho da função de recompensa, reforçando que compreender o ambiente e definir o que deve ser recompensado é a base do aprendizado por reforço.
- Essência do RL: *Um agente de RL interage com seu ambiente e, ao observar as consequências de suas ações, pode aprender a alterar seu próprio comportamento em resposta às recompensas recebidas.* https://arxiv.org/pdf/1708.05866

2. Objeto de Estudo

Investigar como a **organização do ambiente** e o **projeto de recompensas** podem contribuir para melhorar a generalização de modelos de linguagem (LLMs).

Questão central: É possível utilizar feedbacks textuais artificiais, gerados por outro LLM, como informação auxiliar em um processo de aprendizado por reforço em LLMs?

3. Leitura do livro *Reinforcement Learning: An Introduction* (Sutton & Barto)

- **Status:** Não iniciada
- **Decisão:** filtrar a leitura apenas para o **Capítulo 1 (História do RL)**, pois fornece o contexto histórico necessário sem dispersar em tópicos além do foco atual.

4. Leitura de Surveys

- *A Brief Survey of Deep Reinforcement Learning* (2017).
  - Publicado em 2017 (desatualizado para o escopo).
  - Não abrangeu todos os tópicos de interesse definidos.
  - **Ações:** Apesar das limitações, finalizei a leitura e produzi **fichamento** dos seguintes elementos úteis:

- Propriedades gerais de RL.
- Conclusão: o survey será usado como material de apoio histórico.

- Fichamento: A Brief Survey of Deep Reinforcement Learning

- **Segundo Survey:** apresentou o mesmo problema (desatualização e falta de foco).
  - **Ação:** Leitura não iniciada e decisão sobre a realização de uma nova busca de bibliografias.

4. Nova busca bibliográfica mais criteriosa

- Palavras chaves: Reinforcement Learning AND LLM or Large Language Model and Survey
- RLAIF AND RLHF
- Artigo pivot DeepSeek-R1 nas ferramentas indicadas para explorar papers

Como resultado, foi elaborada uma nova seleção de artigos, incluindo um survey de 2025, um estudo sobre o GRPO e outros trabalhos julgados *mais* relevantes. TABELA

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

- Aprofundar a leitura e realizar fichamento do novo survey selecionado “Reinforcement Learning Enhanced LLMs: A Survey” e avaliar sua contribuição para a definição do escopo do meu projeto.

Observação: [caso precise fazer alguma observação, de qualquer “natureza”]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Leituras e Bibliográficas I

Conforme comentado nas seções anteriores, o objeto de estudo visa a intersecção entre RL e LLMs, investigando como a organização do ambiente, o design de recompensas, a escolha de datasets e as estratégias de pipelines podem contribuir para aprimorar a capacidade de modelos de linguagem em determinadas tarefas.

No primeiro levantamento de bibliografias com o foco de introduzir o campo e iniciar nas bases do RL aplicado a LLMs, realizado na semana 1, foram encontrados os seguintes textos: Reinforcement Learning: An Introduction, de Richard Sutton e Andrew Barto[17], que constitui a base teórica clássica sobre RL; Deep Reinforcement Learning: An Overview[18], que apresenta um panorama da evolução do RL e seu contexto atual; e A Brief Survey of Deep Reinforcement Learning[19], um resumo conciso com foco em aplicações modernas.

Durante a semana 2, iniciei a leitura do survey A Brief Survey of Deep Reinforcement Learning e realizei um fichamento do material, porém identifiquei que os dois surveys encontrados não apresentavam suficientemente os avanços recentes de 2024-2025, pois datavam de 2017-2018. Analisei também a necessidade de ler todo o livro de Sutton e Barto, optando por focar apenas no Capítulo 1, que introduz os conceitos fundamentais do Aprendizado por Reforço, suficientes para avançar na etapa, dado que já havia cursado Aprendizado por Reforço como disciplina obrigatória da grade curricular de Inteligência Artificial.

No segundo levantamento bibliográfico, após identificar lacunas na seleção anterior, refinei os critérios de busca para melhor alinhamento com os tópicos de interesse. A estratégia adotada para o levantamento bibliográfico buscou delimitar claramente os tópicos de interesse. Dentre os fundamentos e aplicações recentes em LLMs, destacam-se estudos

[17] SUTTON, Richard S.; BARTO, Andrew G. Reinforcement Learning: An Introduction. 2. ed. Cambridge, MA: The MIT Press, 2015.

[18] ARULKUMARAN, Kai; DEISENROTH, Marc Peter; BRUNDAGE, Miles; BHARATH, Anil Anthony. Deep Reinforcement Learning: An Overview. 2017. arXiv:1701.07274. Disponível em: https://arxiv.org/abs/1701.07274

[19] LI, Yuxi. A Brief Survey of Deep Reinforcement Learning. 2017. arXiv:1708.05866. Disponível em: https://arxiv.org/abs/1708.05866

sobre ajuste fino de modelos de linguagem com RL, particularmente utilizando algoritmos como PPO e GRPO. Como tópicos emergentes, identificou-se a relevância de RLAIF e RLHF para o campo. As palavras-chave utilizadas nas buscas incluíram Reinforcement Learning combinado com LLM ou Large Language Model e Survey. Como artigo pivot, selecionou-se os trabalhos da organização DeepSeek-AI já citados anteriormente.

Tabela de Artigos

A leitura dos artigos selecionados foi realizada nas semanas subsequentes, e ao longo da jornada foram incorporados outros trabalhos considerados relevantes. Abaixo segue a tabela completa, construída interativamente durante todo o período, destacando-se a coluna "Prioridade", que orientou a ordem de leitura. Cabe ressaltar que, embora nem todos os artigos tenham fichamento formal, todos foram submetidos a pelo menos uma leitura preliminar para avaliação de sua pertinência ao estudo.

Título	Resumo	Prioridade
Capítulo 1 – Reinforcement Learning: An Introduction (Sutton & Barto)	Texto clássico que apresenta os fundamentos do aprendizado por reforço: agente, ambiente, política, recompensa, valor e função de valor. Base conceitual essencial para qualquer estudo em RL.	ALTA
A Brief Survey of Deep Reinforcement Learning	Artigo que explora abordagens modernas de aplicação de RL em campos que antes eram inimagináveis.	MÉDIA

DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models	Estudo focado em melhorar raciocínio matemático de LLMs usando técnicas de RL e curadoria de dados. Mostra avanços em benchmarks de matemática e destaca limites e potencial de generalização.	*ARTIGO PIVOT*
DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning	Propõe um pipeline de RL em larga escala para induzir capacidades de raciocínio em LLMs. Mostra que recompensas bem projetadas podem aumentar a coerência e a habilidade de resolver problemas complexos.	*ARTIGO PIVOT*
Reinforcement Learning Enhanced LLMs: A Survey	Survey abrangente sobre como RL tem sido integrado a LLMs, cobrindo métodos como RLHF, RLAIF, PPO/GRPO, bem como aplicações e desafios em generalização, alinhamento e eficiência.	ALTA
A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions	O Survey discute formas de mitigação de alucinação, mas também coloca em evidência as principais causas da alucinação.	ALTA
A Survey of Reinforcement Learning for Large Reasoning Models	Survey extenso sobre os recentes avanços de RL aplicado a LLMs.	MÉDIA

Beyond Two-Stage Training: Cooperative SFT and RL for LLM Reasoning	Discute como integrar SFT e RL de forma cooperativa (em vez de sequencial) para melhorar o raciocínio em LLMs. Defende que essa cooperação acelera a convergência e aumenta a generalização.	MÉDIA
Fine-tune large language models with reinforcement learning from human or AI feedback	Blog da AWS sobre o pipeline de RLAIF.	MÉDIA
UltraFeedback: Boosting Language Models with Scaled AI Feedback	Artigo que propõe uma geração de dataset sintético usando LLMs secundários para avaliar preferências humanas.	MÉDIA
RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback	Artigo que compara RLAIF e RLHF, demonstrando que RLAIF alcança desempenho comparável ao RLHF.	ALTA
Training language models to follow instructions with human feedback	Introduz o InstructGPT e a técnica de alinhamento de LLMs via RLHF.	MÉDIA
Eureka: Human-Level Reward Design via Coding Large Language Models	Usa LLMs secundários para gerar automaticamente funções de recompensa em código executável através de otimização evolutiva.	ALTA

Text2Reward: Reward Shaping with Language Models for Reinforcement Learning	Framework que automatiza a geração de funções de recompensa densas a partir de descrições em linguagem natural.	ALTA
Self-Rewarding Language Models	Propõe modelos que atuam simultaneamente como geradores de resposta e avaliadores de suas próprias saídas, usando o paradigma LLM-as-a-Judge.	ALTA

Fichamento de Citação I[20]

Trecho 1: Um dos principais objetivos do campo da inteligência artificial (IA) é produzir agentes totalmente autônomos que interajam com seus ambientes para aprender comportamentos otimizados, melhorando ao longo do tempo por meio de tentativa e erro. (ARULKUMARAN et al., 2017, p. 1, tradução própria)

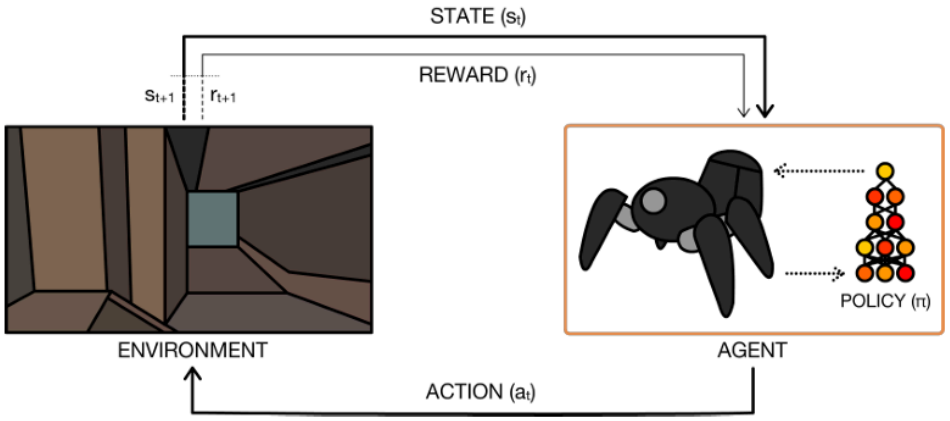
Trecho 2: O aprendizado profundo também acelerou o progresso em RL, com o uso de algoritmos de aprendizado profundo dentro de RL definindo o campo de "aprendizado por reforço profundo" (DRL). O objetivo deste levantamento é cobrir tanto desenvolvimentos seminais quanto recentes em DRL, transmitindo as maneiras inovadoras pelas quais redes neurais podem ser utilizadas para nos aproximar do desenvolvimento de agentes autônomos. (ARULKUMARAN et al., 2017, p. 1, tradução própria)

Trecho 3: A essência do RL é aprender por meio da interação. Um agente de RL interage com seu ambiente e, ao observar as consequências de suas ações, pode aprender a alterar seu próprio comportamento em resposta às recompensas recebidas. Esse paradigma de aprendizado por tentativa e erro tem suas raízes na psicologia behaviorista e é uma das principais bases do RL. A outra influência chave no RL é o controle ótimo, que forneceu os

[20] Idem 19

formalismos matemáticos (notadamente a programação dinâmica) que sustentam o campo. (ARULKUMARAN et al., 2017, p. 2, tradução própria)

Trecho 4: *Figura 2. (ARULKUMARAN et al., 2017, p. 3, tradução própria)*



A imagem acima destaca o ciclo comum do aprendizado de máquina por reforço. De um lado temos o ambiente (environment) que gera o estado atual (st). O agente, por sua vez, recebe esse estado atual e, usando sua política (π), decide qual ação (at) vai tomar, seja andar para frente, virar, etc. O ambiente então reage: mostra um novo estado (st+1) e dá um sinal de recompensa (rt+1) que indica se a ação foi boa ou ruim. Esse ciclo se repete várias vezes. Com o tempo, o agente aprende quais ações valem a pena porque dão mais recompensa.

Trecho 5: *Até agora, introduzimos o formalismo chave usado em RL, o MDP, e mencionamos brevemente alguns desafios em RL. A seguir, iremos distinguir entre diferentes classes de algoritmos de RL. Existem duas abordagens principais para resolver problemas de RL: métodos baseados em funções de valor e métodos baseados em busca de política. Também existe uma abordagem híbrida, ator-crítico, que emprega tanto funções de valor quanto busca de política. Agora, vamos explicar essas abordagens e outros conceitos úteis para resolver problemas de RL. (ARULKUMARAN et al., 2017, p. 4, tradução própria)*

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 17 de set. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Na Semana passada, visando me aprofundar nos conceitos e tópicos de interesse que cercam o aprendizado de máquina por reforço, levantei dois materiais para a leitura.

1. Leitura do Capítulo 1 *Reinforcement Learning: An Introduction* (Sutton & Barto)

- **Status:** Finalizado
- **Destaques:** Apesar do primeiro capítulo não se aprofundar tanto em explicações, ainda assim a leitura foi proveitosa, pois foi possível entender a história do RL, suas influências e propriedades básicas que precisam ser entendidas.
- Como resultado, foi produzido um **fichamento** detalhado destacando as propriedades e trechos do capítulo. Fichamento

2. Leitura Reinforcement Learning Enhanced LLMs: A Survey

- **Status:** Em andamento

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Framework:
DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models - Estudo focado em melhorar raciocínio matemático de LLMs usando técnicas de RL e curadoria de dados. Mostra avanços em benchmarks de matemática e destaca limites e potencial de generalização**.**

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Instituto de Informática (INF) - UFG
Alameda Palmeiras, Quadra D, Câmpus Samambaia
Fone: (62) 3521-1181 / Fax: (62) 3521-1182

Fichamento de Citação II[21]

Trecho 1: *A ideia de que aprendemos interagindo com o nosso ambiente é provavelmente a primeira que nos ocorre quando pensamos sobre a natureza da aprendizagem.[...] A aprendizagem a partir da interação é uma ideia fundamental subjacente a quase todas as teorias de aprendizagem e inteligência. (SUTTON et al., 2015, p. 15, tradução própria)*

Trecho 2: *Os problemas de aprendizagem por reforço envolvem aprender o que fazer – como mapear situações para ações – de modo a maximizar um sinal de recompensa numérica.[...] Além disso, o aprendiz não é instruído sobre quais ações tomar, como em muitas formas de aprendizagem de máquina, mas deve descobrir quais ações produzem a maior recompensa experimentando-as. Nos casos mais interessantes e desafiadores, as ações podem afetar não apenas a recompensa imediata, mas também a próxima situação e, através dela, todas as recompensas subsequentes. (SUTTON et al., 2015, p. 16, tradução própria)*

Trecho 3: *Claramente, tal agente deve ser capaz de sentir o estado do ambiente até certo ponto e deve ser capaz de tomar ações que afetem o estado. O agente também deve ter um objetivo ou objetivos relacionados com o estado do ambiente. A formulação destina-se a incluir apenas estes três aspetos – sensação, ação e objetivo – nas suas formas mais simples possíveis, sem trivializar nenhum deles. (SUTTON et al., 2015, p. 16, tradução própria)*

Trecho 4: *A aprendizagem supervisionada é a aprendizagem a partir de um conjunto de treino de exemplos rotulados fornecidos por um supervisor externo conhecedor. Cada exemplo é uma descrição de uma situação juntamente com uma especificação – o rótulo – da ação correta que o sistema deve tomar para essa situação, que é frequentemente identificar uma categoria à qual a situação pertence. O objetivo deste tipo de aprendizagem é que o sistema extrapole, ou generalize, as suas respostas para que atue corretamente em situações não presentes no conjunto de treino. Este é um tipo importante de aprendizagem,*

[21] Idem 17

mas por si só não é adequado para a aprendizagem a partir da interação. (SUTTON et al., 2015, p. 16, tradução própria)

Trecho 5: *A aprendizagem por reforço também é diferente do que os investigadores de aprendizagem de máquina chamam de aprendizagem não supervisionada, que tipicamente se trata de encontrar estruturas ocultas em coleções de dados não rotulados. [...] Consideramos, portanto, a aprendizagem por reforço como um terceiro paradigma de aprendizagem de máquina, ao lado da aprendizagem supervisionada, da aprendizagem não supervisionada e, talvez, de outros paradigmas também. (SUTTON et al., 2015, p. 17, tradução própria)*

Trecho 6: *Um dos desafios que surgem na aprendizagem por reforço, e não em outros tipos de aprendizagem, é o trade-off entre "exploration" e "exploitation". Para obter uma recompensa, um agente de aprendizagem por reforço deve preferir ações que já experimentou no passado e que considerou eficazes na produção de recompensa. Mas para descobrir tais ações, ele tem que experimentar ações que não selecionou antes. (SUTTON et al., 2015, p. 17, tradução própria)*

Trecho 7: *Outra característica fundamental da aprendizagem por reforço é que ela considera explicitamente todo o problema de um agente orientado por objetivos interagindo com um ambiente incerto. Isso contrasta com muitas abordagens que consideram subproblemas sem abordar como elas poderiam se encaixar em um quadro maior. (SUTTON et al., 2015, p. 17, tradução própria)*

Trecho 8: *Finalmente, a aprendizagem por reforço também faz parte de uma tendência maior na inteligência artificial de retorno a princípios gerais simples. Desde o final da década de 1960, muitos pesquisadores de inteligência artificial presumiram que não havia princípios gerais a serem descobertos, que a inteligência era, em vez disso, devido à posse de um vasto número de truques, procedimentos e heurísticas de propósito especial. Às vezes, dizia-se que se pudéssemos apenas colocar fatos relevantes suficientes em uma máquina, digamos um milhão, ou um bilhão, então ela se tornaria inteligente. (SUTTON et al., 2015, p. 18, tradução própria)*

Trecho 9: *Phil prepara o seu pequeno-almoço. Examinada de perto, até esta atividade aparentemente mundana revela uma complexa teia de comportamento condicional e relações interligadas de objetivo-subobjetivo: caminhar até ao armário, abri-lo, selecionar uma caixa de cereais, depois estender a mão, agarrar e retirar a caixa. [...] Todos envolvem a interação entre um agente ativo de tomada de decisão e o seu ambiente, dentro do qual o agente procura atingir um objetivo apesar da incerteza sobre o seu ambiente. (SUTTON et al., 2015, p. 19, tradução própria)*

Trecho 10: *Além do agente e do ambiente, podem-se identificar quatro subelementos principais de um sistema de aprendizagem por reforço: uma política, um sinal de recompensa, uma função de valor e, opcionalmente, um modelo do ambiente. (SUTTON et al., 2015, p. 21, tradução própria)*

Trecho 11: *Uma política define a forma de comportamento do agente de aprendizagem num dado momento. Grosso modo, uma política é um mapeamento de estados percebidos do ambiente para ações a serem tomadas nesses estados. (SUTTON et al., 2015, p. 21, tradução própria)*

Trecho 12: *Um sinal de recompensa define o objetivo num problema de aprendizagem por reforço. A cada passo do tempo, o ambiente envia ao agente de aprendizagem por reforço um único número, uma recompensa. (SUTTON et al., 2015, p. 21, tradução própria)*

Trecho 13: *Enquanto o sinal de recompensa indica o que é bom num sentido imediato, uma função de valor especifica o que é bom a longo prazo. Grosso modo, o valor de um estado é a quantidade total de recompensa que um agente pode esperar acumular no futuro, a partir desse estado. (SUTTON et al., 2015, p. 22, tradução própria)*

APÊNDICE 2

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: **25 de set. de 2025**

Participantes da Entrega **[matriculados em Residência em IA]:**

Lisandra Cristina de Moura Menezes

Entrega: **[descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]**

Na Semana anterior, desenvolvi um fichamento sobre os conceitos iniciais de RL e como planejamento estava a finalização de duas leituras: "*Reinforcement Learning Enhanced LLMs: A Survey*" e "*DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models*."

1. **Reinforcement Learning Enhanced LLMs: A Survey**

- Status: Finalizada
- Destaques: Foi realizado um fichamento do survey em que destacamos:
  - Estrutura base do processo de aprimoramento de LLMs por RL (modelo de recompensa, Ajuste Fino baseado em preferência e Otimização de Política); Uma estrutura base é definida de forma mais clara no artigo a seguir.
  - A distinção entre RLHF e RLAIF reside na origem do feedback, mas existe uma ampla gama de abordagens para RLAIF que quero explorar, tais como: destilação de datasets, prompting como função de recompensa e auto-recompensa.

2. **DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models**

- Status: Finalizada
- Destaques: Embora o artigo do DeepSeekMath tenha o objetivo de apresentar o modelo treinado usando o GRPO, um algoritmo simplificado do PPO, o artigo trouxe algumas definições que foram essenciais para o meu entendimento do aprimoramento de LLMs. Desenvolvi um fichamento limitado à seção 4 e 5.
  - O PPO é computacionalmente complexo porque, além da função de recompensa que avalia as respostas do LLM (ator), exige um **modelo crítico** que estima o valor esperado dessa recompensa. A combinação entre recompensa real e estimativa do crítico produz a **vantagem**, usada para atualizar o ator.
  - O GRPO elimina a necessidade do modelo crítico e em vez de prever o valor esperado da recompensa, ele gera **N respostas para o mesmo prompt**, calcula a recompensa de cada uma e depois normaliza. Assim, cada resposta é avaliada em relação às demais, dispensando a estimativa do crítico e reduzindo a complexidade.
  - Uma contribuição que me ajudou a entender os diferentes métodos de aprimoramento de LLMs foi os seguintes componentes:
    - Fonte de dados
    - Função de recompensa

- Algoritmo que processa os elementos anteriores

Montei uma tabela com SFT, PPO, DPO e GRPO usando esses componentes para diferenciar cada método. Com esses componentes conseguimos classificar se o método é on-line ou off-line, como é a função de recompensa e como é a lógica da atualização dos pesos no modelo de linguagem. Tabela de Métodos

Documentação Completa

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Para a próxima Semana, pretendo focar e refinar o mapeamento de abordagens identificadas de RLAIF.

1. Abordagens mapeadas
   - Destilação de Feedback de IA para Treinar Modelo de Recompensa.
   - Prompting de LLMs como Função de Recompensa.
   - Auto-recompensa
2. Buscar variações de AI feedback não exploradas no Survey
3. Catalogar limitações do uso de AI Feedback
4. Para além de refinamentos metodológicos, pretendo iniciar uma exploração de frameworks que suportam RLAIF para identificar quais abordagens são suportadas.
   - ART- https://github.com/OpenPipe/ART
   - Atropos - https://github.com/NousResearch/Atropos
   - Explorar outros frameworks

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Paradigma RL

Uma das contribuições mais valiosas identificadas na literatura é o framework teórico que unifica diferentes métodos de treinamento sob uma perspectiva comum. Shao et al. [22] demonstram que técnicas aparentemente distintas como SFT, RFT, DPO, PPO e GRPO podem ser compreendidas através de três componentes fundamentais: a fonte de dados, a função de recompensa e o algoritmo que processa esses elementos em coeficientes de gradiente.

Esta unificação revela insights importantes sobre as diferenças entre métodos online e offline. O treinamento online, onde os dados são amostrados do modelo atual durante o treinamento, demonstra superioridade clara sobre abordagens offline que dependem de dados estáticos do modelo SFT inicial. Os experimentos dos autores mostram que o Online RFT supera consistentemente o RFT tradicional, especialmente em estágios posteriores do treinamento, quando a divergência entre o modelo atual e o modelo inicial se torna mais pronunciada.

Paralelamente, Ouyang et al. descrevem um pipeline de treinamento focado em RLHF que inicia com a coleta de dados de demonstração (por exemplo, prompts acompanhados de respostas esperadas) seguida de uma etapa de SFT. Essa etapa garante que o modelo se ajuste à distribuição desejada antes do treinamento por reforço. Em seguida, realiza-se uma coleta de dados de preferência, na qual o modelo gera múltiplas alternativas para cada prompt e os avaliadores humanos classificam as respostas. Um modelo de recompensa é então treinado para emular esse classificador humano durante a fase de alinhamento via RL. Com os dados coletados e o modelo de recompensa treinado, aplica-se o algoritmo de RL para ajustar a política do LLM com base nas recompensas geradas.

Vale ressaltar que o pipeline descrito por Ouyang et al. é específico para preferências, usando modelos de recompensa e RLHF. A seguir, propomos um pipeline mais generalista,

[22] Idem 5

capaz de atender a diferentes abordagens identificadas na literatura, incluindo RLAIF e suas variantes.

Etapa 0: Contexto Inicial (Opcional mas Recomendado)

A maioria dos pipelines de RL inicia com um modelo que já passou por algum ajuste supervisionado para garantir que as gerações estejam dentro de uma distribuição desejável. Caso o objetivo seja treinar do zero (como no DeepSeek-R1-Zero), esta etapa pode ser omitida. É imprescindível, todavia, testar se o modelo que você está querendo usar possui conhecimento sobre o assunto que deseja, como argumenta WANG et al., Out-of-distribution (OOD) podem apresentar um problema para o modelo.

Etapa 1: Definição do Objetivo e Comportamento Desejado

O primeiro passo consiste em definir claramente o que se deseja otimizar no modelo, ou seja, qual comportamento deve ser incentivado ou corrigido. Existe atualmente uma gama de aplicações possíveis, como raciocínio lógico-matemático, alinhamento a preferências humanas, redução de alucinações, melhoria de clareza.

Decisões nesta etapa:

1. Qual habilidade ou comportamento será priorizado?
2. Haverá múltiplos objetivos?
3. Como esses objetivos serão traduzidos em critérios mensuráveis?

Esta definição orientará todas as etapas subsequentes, especialmente na construção da função de recompensa.

Etapa 2: Coleta e Preparação dos Dados (Prompts)

Reunir instruções (prompts) que o modelo deverá responder durante o treinamento.

Fontes possíveis:

1. Datasets públicos (ReClor, UltraFeedback)

2. Gerações automáticas (self-instruct)
3. Interações reais de usuários

Essa coleta pode ser feita antes do treino (offline) ou durante o treino (online), dependendo do algoritmo escolhido na Etapa 5.

Etapa 3: Geração de Respostas (Outputs)

Nesta etapa, o modelo gera uma ou mais respostas para cada prompt. A quantidade e forma de geração dependem da estratégia escolhida:

Geração única por prompt:

1. Usada em abordagens de recompensa direta (sem comparação)
2. Métodos compatíveis: reward modeling direto, métricas automáticas

Geração múltipla por prompt:

1. Quantidade típica: 2-8 respostas por prompt (DPO usa 2; GRPO pode usar 64+)
2. Métodos compatíveis: DPO (pares chosen/rejected), PPO, GRPO

Etapa 4: Definição ou Geração da Função de Recompensa

Esta é a etapa central do RL aplicado a LLMs. Aqui, define-se como medir o valor de uma resposta que será considerado uma boa ação.

RLHF (Reinforcement Learning from Human Feedback)

1. Avaliadores humanos classificam, comparam ou pontuam respostas
2. Dados de preferência são coletados (ex: "resposta A melhor que B")
3. Um modelo de recompensa (Reward Model) é treinado para emular o julgamento humano
4. Durante RL, o RM avalia automaticamente novas gerações

RLAIF (Reinforcement Learning from AI Feedback)

Tem como o objetivo substituir ou complementar avaliações humanas, podemos substituir completamente como o humano por um AI no RLHF descrito anteriormente, mas há outras maneiras de usar AI no processo de feedback.

Destilação de Feedback de IA

1. Um LLM avançado (ex: GPT-4) anota milhares de exemplos automaticamente
2. Gera preferências artificiais (chosen vs rejected) ou notas numéricas
3. Um modelo de recompensa é treinado com esses dados destilados
4. Durante RL, o RM avalia automaticamente novas gerações

Prompting Rewarding (LLM-as-a-Judge)

1. Durante o treinamento, cada resposta gerada é enviada via API para um LLM avaliador
2. O LLM retorna uma nota numérica ou avaliação textual baseada em prompt estruturado
3. Essa nota é usada diretamente como recompensa (sem treinar RM intermediário)

Self-Rewarding

1. O próprio modelo em treinamento atua como seu avaliador
2. Gera respostas candidatas e as avalia (LLM-as-a-Judge interno)
3. Usa essas autoavaliações como sinal de treinamento

Recompensas Diretas (Sem Modelo de Recompensa)

1. Acurácia: verifica se resposta final está correta (ex: problemas matemáticos)
2. Comprimento: penaliza respostas muito longas ou curtas
3. Formatação: recompensa uso correto de tags, estrutura
4. Métricas linguísticas: perplexidade, diversidade lexical
5. Verificadores externos: executar código gerado, consultar bases de conhecimento

Etapa 5: Otimização da Política

Aplicar algoritmo de RL para ajustar os pesos do modelo com base nas recompensas. Na Tabela de Métodos, apresento uma discussão mais aprofundada sobre os métodos e seus respectivos benefícios.

Etapa 6: Avaliação

Verificar se o modelo melhorou no objetivo definido usando métricas como Pass@k, acurácia em benchmarks, win rate, análise qualitativa de respostas.

Fichamento de Citação III[23]

No aprendizado por reforço (RL), existem seis componentes principais: agente, ambiente, estado, ação, recompensa e política. Para aplicar RL para ajuste fino de grandes modelos de linguagem (LLMs), o primeiro passo é mapear esses componentes para o framework LLM. LLMs são altamente proficientes em prever o próximo token, onde recebem uma sequência de tokens como entrada e preveem o próximo token com base no contexto dado. Do ponto de vista do RL, podemos ver o LLM em si como a política. A sequência textual atual representa o estado e, com base nesse estado, o LLM gera uma ação — o próximo token. Essa ação atualiza o estado, criando um novo estado que incorpora o token recém-adicionado. Após gerar uma sequência textual completa, uma recompensa é determinada avaliando a qualidade da produção do LLM usando um modelo de recompensa pré-treinado. (WANG et al., 2025, p. 4)

LLMs populares recentes, com fortes capacidades, quase todos utilizam aprendizado por reforço (RL) para aprimorar ainda mais seu desempenho durante o processo pós-treinamento. Os métodos RL adotados por esses modelos podem ser tipicamente divididos em duas linhas principais: 1. Abordagens tradicionais de RL, como Aprendizado por Reforço a partir de Feedback Humano (RLHF) e Aprendizado por Reforço a partir de Feedback de IA (RLAIF). (WANG et al., 2025, p. 4)

[23] Idem 4

Aprendizado por reforço a partir do feedback humano (RLHF) é uma abordagem de treinamento que combina aprendizado por reforço (RL) com feedback humano para alinhar os LLMs com os valores, preferências e expectativas humanas. (WANG et al., 2025, p. 11, tradução própria)

Uma vez treinado o modelo de recompensa, ele é usado para guiar o ajuste fino do LLM original por meio do aprendizado por reforço. (WANG et al., 2025, p. 12, tradução própria)

O aprendizado por reforço a partir do feedback de IA (RLAIF) serve como uma alternativa promissora ou complemento ao RLHF que aproveita sistemas de IA — frequentemente LLMs mais potentes ou especializados (por exemplo, GPT-4 OpenAI (2024a))—para fornecer feedback sobre os resultados do LLM sendo treinado. (WANG et al., 2025, p. 12, tradução própria)

Além dos dados coletados manualmente, a destilação de conjuntos de dados de LLMs pré-treinados apresenta uma alternativa eficiente. Ao aproveitar os resultados de LLMs poderosos como o GPT-4, os pesquisadores podem construir uma ponte entre a curadoria manual e a avaliação autônoma. (WANG et al., 2025, p. 12, tradução própria)

À medida que o treinamento com modelos de recompensa se torna mais sofisticado, uma progressão natural é empregar os próprios LLMs como avaliadores no ciclo do aprendizado por reforço. (WANG et al., 2025, p. 13, tradução própria)

O mecanismo auto-recompensador permite que o LLM avalie e refine autonomamente seu próprio desempenho, abordando as limitações de custo, escalabilidade e adaptabilidade dos métodos RL existentes. (WANG et al., 2025, p. 16, tradução própria)

Fichamento de Citação IV[24]

Proximal Policy Optimization (PPO) (Schulman et al., 2017) is an actor-critic RL algorithm that is widely used in the RL fine-tuning stage of LLMs (Ouyang et al., 2022). (SHAO et al., 2025, p. 11)

[24] Idem 5

As the value function employed in PPO is typically another model of comparable size as the policy model, it brings a substantial memory and computational burden. (SHAO et al., 2025, p. 13)

[...] we propose Group Relative Policy Optimization (GRPO), which obviates the need for additional value function approximation as in PPO, andinstead uses the average reward of multiple sampled outputs, produced in response to the same question, as the baseline. (SHAO et al., 2025, p. 13)

Formally, for each question q, a group of outputs {$o1$,$o2$,$\cdots$,oG} are sampled from the old policy model $\pi\theta old$. A reward model is then used to score the outputs, yielding G rewards r = {$r1$,$r2$,$\cdot\ \cdot\ \cdot$,rG} correspondingly. Subsequently, these rewards are normalized by subtracting the group average anddividing bythegroupstandarddeviation. Outcome supervision provides the normalized reward at the end of each output oi and sets the advantages $\hat{}Ai,t$ of all tokens in the output as the normalized reward, i.e., Âi,t =ri = ri−mean(r) /std(r) maximizing the objective defined in equation. (SHAO et al., 2025, p. 14)

In this paper, we conduct reinforcement learning based on a subset of instruction tuning data, and it achieves significant performance enhancement upon the instruction tuning model. To further explain why reinforcement learning works. We evaluate the Pass@K and Maj@K accuracy of the Instruct and RL models on two benchmarks. As shown in Figure 7, RL enhances Maj@K's performance but not Pass@K. These findings indicate that RL enhances the model's overall performance by rendering the output distribution more robust, in other words, it seems that the improvement is attributed to boosting the correct response from TopK rather than the enhancement of fundamental capabilities. Similarly, (Wang et al., 2023a) identified a misalignment problem in reasoning tasks within the SFT model, showing that the reasoning performance of SFT models can be improved through a series of preference alignment strategies (Song et al., 2023; Wang et al., 2023a; Yuan et al., 2023b). (SHAO et al., 2025, p. 21)

Tabela de Métodos

Usando a classificação de *SHAO et al.*[25], a proposta dessa tabela é definir se o método usa uma abordagem on-line ou off-line (isto é, como é a fonte de dados), a função de recompensa e como funciona o algoritmo que processa o gradiente.

MÉTODO	DESCRIÇÃO	USO
SFT	O Ajuste Fino Supervisionado (SFT) é o processo de aprimoramento de um LLM já pré-treinado para que ele aprenda a responder de acordo com um comportamento desejado. Para isso, utiliza-se um conjunto de dados fixo e anotado, normalmente pares de perguntas e respostas esperadas. Esse método é considerado off-line, porque o modelo não interage em tempo real para receber feedback, apenas aprende a partir deste dataset estático. Durante o treinamento, o modelo gera probabilidades para cada palavra (token) da resposta, e a função de perda compara essas probabilidades previstas com as respostas ideais do dataset. Quanto mais próxima a previsão estiver do rótulo correto, menor será a perda. Assim, uma loss baixa indica que o modelo está reproduzindo com mais fidelidade o comportamento esperado, ou seja, está aprendendo a responder no formato desejado. No SFT, usamos um novo dataset supervisionado e recalculamos a loss para nos aproximarmos dos exemplos fornecidos. Para atualizar os pesos, seguimos o mesmo processo do treinamento de uma rede neural.	O SFT geralmente é sugerido para alteração de um comportamento do modelo, isto é, em caso que queremos forçar um determinado tipo de resposta ou um estilo de escrita, por exemplo, para criação de petições iniciais e contestações com linguagem jurídica. Além disso, *WANG et al.*[26] cita que alguns modelos passaram por uma etapa inicial de SFT para aumentar a probabilidade de determinadas respostas. Também é útil em cenários que queremos destilar conhecimentos de modelos maiores em modelos menores, como é o caso dos modelos destilados em DeepSeek-AI.
PPO	O PPO é um algoritmo actor-critic de RL onde o Ator (o LLM) gera respostas e o Crítico estima o valor esperado dessas	O PPO faz parte da família de métodos de alinhamento, que têm como objetivo

[25] Idem 5
[26] Idem 4

	respostas. Um modelo de recompensa ou função de recompensa avalia a qualidade final de cada resposta gerada. A diferença entre a recompensa real e a estimativa do Crítico resulta na vantagem, que guia o ajuste do modelo. Para evitar mudanças drásticas, o PPO adiciona uma penalização KL que limita o quanto o modelo pode se afastar de sua política original. Essa combinação, maximizar recompensa e regularizar mudanças, mantém o modelo estável durante o treinamento. É uma abordagem on-line porque gera dados durante o próprio treinamento. A função de recompensa pode ser personalizada conforme o objetivo: avaliações humanas, critérios automáticos ou outro LLM como juiz.	ajustar o comportamento de LLMs a partir de feedback humano ou de outros modelos. Diferentemente de abordagens puramente supervisionadas (como o SFT), o PPO é particularmente adequado para tarefas abertas, nas quais não existe uma resposta única correta. Isso acontece porque o PPO permite que o pesquisador defina funções de recompensa personalizadas ou modelos de recompensa, refletindo critérios como clareza, criatividade, utilidade ou segurança, de acordo com o problema específico. Para *SHAO et al.*[27], etapas de alinhamento com RL estão corrigindo um erro crítico do SFT, na qual o modelo possui o conhecimento necessário para gerar respostas corretas, mas sua distribuição de probabilidade não prioriza adequadamente essas respostas.
DPO	O Direct Preference Optimization (DPO) é uma técnica para alinhar LLMs com preferências que simplifica o processo em relação ao PPO. Em vez de usar pares de pergunta e resposta esperada, o DPO trabalha com pares de respostas comparadas: uma resposta preferida (chosen) e uma resposta rejeitada (rejected) para a mesma instrução. Esses pares	O DPO é uma técnica de alinhamento para LLMs baseada em preferências. Seu objetivo é aproximar o comportamento do modelo das respostas que os humanos preferem, treinando diretamente em pares chosen vs rejected.

[27] Idem 5

	provêm de momentos em que usuários (ou anotadores) escolhem qual saída do modelo é melhor, gerando um dataset de preferências. A grande diferença em relação ao PPO é que o DPO não precisa treinar um modelo de recompensa nem executar um algoritmo de RL complexo: a função de recompensa é incorporada diretamente à função de perda (loss), permitindo ajuste de pesos sem loop de reforço. Na prática, o algoritmo aumenta a probabilidade de gerar a resposta chosen em relação à rejected. Se o modelo já favorece a resposta preferida, a loss é baixa; se ainda favorece a rejeitada, a loss é alta, forçando ajustes nos pesos. Como no SFT, o DPO é off-line, pois utiliza um dataset fixo de preferências. A diferença é que, em vez de imitar respostas corretas, ele aprende a preferir uma saída em relação à outra.	Funciona muito bem quando há comparações diretas de preferências, mas é menos indicado para problemas abertos que exigem funções de recompensa complexas ou compostas (ex.: segurança + criatividade + factualidade), sendo menos generalista que o PPO nesse aspecto.
GRPO	O Group Relative Policy Optimization (GRPO) é uma variante do PPO simplificada que elimina o modelo crítico e usa estimativa de vantagem baseada em grupo para reduzir a sobrecarga computacional, mas mantendo as demais características do PPO. Essa abordagem também se caracteriza como on-line, e usa os mesmos formatos de função de recompensa e algoritmos de ajuste, só mudando mesmo a forma de gerar a estimativa de vantagem.	O GRPO faz parte da família de métodos de alinhamento que têm como objetivo ajustar o comportamento de LLMs a partir de feedback humano ou de outros modelos. Como variante do PPO, o GRPO tem os mesmos usos que sua variante.

APÊNDICE 3

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 2 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Na Semana anterior, iniciei a construção de uma tabela com algoritmos de aperfeiçoamento de modelos de linguagem que têm como objetivo refinar a probabilidade de geração de tokens alinhados às preferências humanas.

Como registrado, algumas abordagens principais de RLAIF já haviam sido mapeadas, foi realizado nesta Semana a trilha de: **Destilação de datasets de feedback de IA para treinamento de modelos de recompensa**.

1. Planejei a trilha de RLAIF, estruturada inicialmente em torno das três abordagens, mas com o objetivo de identificar e incluir outras variações.
2. Estruturei uma série de tabelas comparativas para consolidar o meu conhecimento:
    a. **Tabela 1** – Abordagens: resumo das principais técnicas, seus objetivos, funcionamento e limitações.
    b. **Tabela 2** – Algoritmos: descrição técnica dos métodos de aperfeiçoamento aplicados em LLMs.
    c. **Tabela 3** – Ferramentas: comparação entre bibliotecas, frameworks e datasets que dão suporte à implementação de RLAIF.
3. Iniciei os estudos aprofundados sobre as abordagens de Destilação em que li papers que descreviam pipelines de destilação desses datasets.
    a. **RLAIF vs. RLHF:** compara humanos e IA como anotadores. O processo com IA é análogo ao humano, mas a escolha é feita pela probabilidade de o modelo preferir uma resposta. Conclui-se que tanto RLHF quanto RLAIF superam STF e melhoram significativamente o modelo.
    b. **UltraFeedback:** propõe um dataset robusto com anotações detalhadas (veracidade, honestidade, utilidade, seguimento do prompt e crítica). As respostas foram geradas por diferentes modelos e avaliadas por outro LLM.
4. Iniciei a criação de tutoriais e guias em um repositório, na tentativa de consolidar as leituras feitas em uma prática que posso consultar sempre que for necessário. *(A ideia não é só criar, mas usar o que tem pronto, nesse primeiro momento não encontrei guias prontos)*

Documentação completa

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Pendências da Destilação:

- Criar um pipeline de treinamento de modelos de recompensa usando uma amostra do **UltraFeedback**

Continuidade na trilha:

- Prompting de LLMs como Função de Recompensa (preencher tabelas, continuar guias/tutoriais, leituras)

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Trilha RLAIF I - Destilação de Dataset

1. Objetivo

A destilação de datasets, no contexto de RLHF e RALF, consiste em transformar julgamentos de qualidade de humanos ou de LLMs mais fortes em dados estruturados que permitam treinar outro modelo para se alinhar melhor a determinados objetivos. O propósito central é produzir conjuntos de preferências ou anotações que substituem ou complementam o feedback humano, possibilitando tanto o treinamento de modelos de recompensa quanto a aplicação direta de algoritmos de reforço, como PPO ou DPO.

Cada tarefa exige um tipo específico de anotação. Por exemplo, se a meta é reduzir toxicidade, utiliza-se um conjunto de perguntas e respostas comuns e solicita-se a um LLM que classifique o nível de toxicidade, gerando dados artificiais que orientam o modelo-alvo a evitar respostas nocivas. Esse princípio é ilustrado pelo UltraFeedback, que coleta múltiplas respostas de diferentes modelos e utiliza o GPT-4 como juiz para atribuir notas e críticas em várias dimensões, formando um dataset sintético robusto apto a treinar modelos de recompensa mais granulares.

Um ponto central é o uso de LLMs como avaliadores. Como esses modelos já passaram por RLHF, seus julgamentos apresentam alta correlação com avaliações humanas. Isso traz vantagens como escalabilidade, baixo custo e alinhamento com valores humanos, mas também limitações, pois eventuais vieses e falhas presentes nesses modelos são igualmente propagadas. Assim, a destilação de datasets se consolida como uma estratégia para alinhar LLMs a diferentes objetivos de qualidade, com anotações ajustadas ao método de treinamento e ao propósito específico.

Como etapa adicional da trilha, foi desenvolvido um repositório com versões simplificadas das implementações apresentadas nos artigos estudados[28] e tabelas comparativas das leituras realizadas juntamente com materiais que poderiam ser úteis em etapas de experimentação. As tabelas comparativas podem ser consultadas na Trilha III.

[28] https://github.com/LisandraMoura/Track_RLAIF

2. Fichamento Resumo

RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback[29]

No artigo, são utilizados os datasets Anthropic HH (Helpful & Harmless)[30], TL;DR Summarization from Human Feedback[31] e OpenAI LM Human Preferences[32], todos compostos por pares de respostas anotados manualmente por humanos. Esses conjuntos capturam preferências humanas e sustentam o treinamento de modelos via RL, caracterizando o paradigma RLHF. No RLAIF, o artigo preserva os mesmos prompts e estruturas de tarefa, mas substitui o avaliador humano por um LLM que atua como anotador de preferências. Assim, gera-se um dataset paralelo com preferências artificiais, alinhado aos datasets originais, porém produzido de forma automática e escalável.

O processo de destilação ocorre da seguinte forma: um LLM secundário é usado como "juiz" para escolher entre duas respostas. O prompt possui quatro partes: instruções iniciais, exemplos opcionais (few-shot), o par de respostas e uma pergunta final solicitando a escolha. O modelo produz probabilidades para as opções "1" e "2", que são normalizadas para formar uma distribuição de preferência. Para mitigar viés posicional, a ordem das respostas é invertida, e as duas avaliações são agregadas. Essas preferências geram *soft labels*, usados para treinar um modelo de recompensa (RM). O RM aprende a imitar as escolhas do LLM e, posteriormente, guia o treinamento por reforço da política. O artigo também propõe uma variação, *direct-RLAIF*, em que o LLM atribui notas diretas (1 a 10), usadas como sinal de recompensa, dispensando o RM intermediário.

[29] LEE, Harrison; et al. *RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback*. 2023. arXiv preprint arXiv:2309.00267. Disponível em: https://arxiv.org/abs/2309.00267.
[30] ANTHROPIC. *Helpful and Harmless (HH) Dataset*. [S.l.], 2022. Disponível em: https://huggingface.co/datasets/Anthropic/hh-rlhf.
[31] OPENAI. TL;DR: Summarization from Human Feedback. [S.l.], 2020. Disponível em: https://openaipublic.blob.core.windows.net/summarize-from-feedback/website/index.html#/tldr
[32] OPENAI. *LM Human Preferences*. [S.l.], 2021. Disponível em: https://github.com/openai/lm-human-preferences.

UltraFeedback: Boosting Language Models with Scaled AI Feedback[33]

O estudo visa escalar datasets de feedback substituindo anotadores humanos por LLMs avançados, como o GPT-4. O uso de IA reduz custos e facilita a expansão dos dados. O pipeline consiste em:

1. Coleta das instruções: foram usadas instruções do UltraChat (mais de 1,4 milhão), das quais cerca de 64 mil únicas foram selecionadas para anotação.
2. Geração de múltiplas respostas: para cada instrução, coletaram quatro respostas de diferentes modelos (incluindo modelos abertos como LLaMA-2, Falcon, Vicuna, entre outros). Isso gera um conjunto diverso de saídas candidatas.
3. Avaliação com GPT-4: cada grupo de quatro respostas é avaliado pelo GPT-4, que fornece notas numéricas em quatro dimensões: *seguir instruções, veracidade, honestidade e utilidade*, além de Feedback textual crítico, apontando prós e contras de cada resposta.
4. Construção do dataset final – o resultado é um dataset de mais de 340 mil de preferências anotados por IA, com a pontuação em cada nível avaliado.

Os dados do UltraFeedback foram usados para treinar três tipos de modelos:

- UltraRM: um modelo de recompensa.
- UltraLM-13B-PPO: um LLM alinhado com PPO, usando o UltraRM como sinal de recompensa.
- UltraCM: um modelo treinado para gerar críticas textuais automáticas, imitando os feedbacks fornecidos pelo GPT-4.

5. Reflexão crítica

A destilação de datasets permite treinar modelos de recompensa capazes de otimizar objetivos complexos ao nível da sequência, algo inalcançável pelo SFT tradicional. No entanto, o pipeline é custoso: envolve múltiplas etapas, consumo elevado de GPU e

[33] CUI, Ganqu et al. UltraFeedback: Boosting Language Models with Scaled AI Feedback. 2023. arXiv preprint arXiv:2310.01377. Disponível em: https://arxiv.org/abs/2310.01377.

complexidade operacional, frequentemente para alcançar resultados próximos aos obtidos por métodos mais diretos, como *reward prompting*.

Os resultados apresentados no artigo mostram que RLAIF e RLHF têm desempenho estatisticamente semelhante em tarefas de sumarização e diálogo útil (71% vs 73% em sumarização; 63% vs 64% em diálogo útil), ambos superando significativamente o SFT. Em diálogo inofensivo, o RLAIF apresenta maior taxa de indiferença, mas também excede o SFT. Esses achados sugerem que, embora a destilação via modelos secundários ofereça maior estabilidade e controle, o ganho marginal frente ao custo pode ser limitado. Em síntese, a destilação é uma ferramenta poderosa para consolidar feedback em larga escala e sistematizar preferências, mas sua eficiência prática merece reavaliação, especialmente diante de alternativas mais simples que produzem resultados comparáveis.

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 9 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Utilizando RL para alinhamento de LLMs

Nas primeiras Semanas, me dediquei a compreender a área de Aprendizado por Reforço (RL) e iniciei uma aprofundamento em RL aplicado a LLMs, como resultado produzi materiais de apoio. Neste estudo, identifiquei duas vertentes principais: **RLHF** e **RLAIF**.

Para discutir o trabalho desta Semana, é útil esclarecer as **siglas** e o framework de alinhamento de LLMs com Reforço Learning (RL), baseado no artigo *Training language models to follow instructions with human feedback*.

No **RLHF (Reinforcement Learning from Human Feedback)**, o alinhamento ocorre a partir de feedback humano.

No **RLAIF (Reinforcement Learning from AI Feedback)**, ao contrário, usa-se uma IA generativa para gerar o feedback que servirá como aprendizado.

Nas últimas duas Semana trouxe uma classificação sobre RLAIF e trouxe minhas leituras sobre a "Destilação de datasets", que na prática é quando um LLM assume esse papel de avaliador, atribuindo notas ou preferências automáticas às respostas. Assim, o modelo aprende a se alinhar com base em feedback de IA em vez de humanos.

Para as próximas Semanas, ficaram então **outras duas classificações identificadas**:

- *Prompting de LLMs como Função de Recompensa*: Como usar LLMs para **gerar recompensas**?
- *Self-reward*: É possível usar o **mesmo LLM que está sendo aprimorado** para se autoavaliar?

Entregas:

1. **Modelo de Recompensa**

Usa-se um modelo pré-treinado e substitui a camada final de geração de texto por uma camada linear. Com os dados de preferência, por exemplo, ele aprende a dar uma pontuação escalar e durante a fase de alinhamento, o modelo de recompensa é usado como "função de recompensa" para as respostas do LLM.
No artigo Ultra Feedback - Devido às limitações de hardware, não foi possível carregar o modelo

openbmb/UltraRM-13b, optando-se por prosseguir os testes sem o uso simultâneo de dois LLMs.

2. **Prompting de LLMs como Função de Recompensa**

Leitura de três artigos que exploram como LLMs podem atuar como avaliadores automáticos, atribuindo recompensas diretas às respostas geradas durante o treinamento. Nesses pipelines de aprendizado por reforço, o próprio LLM analisa e pontua as saídas de outro modelo, substituindo a função de recompensa tradicional e permitindo um processo de alinhamento sem anotações humanas.

3. **Self-reward**

Investigar e desenvolver métodos que permitam que modelos de linguagem aprendam a se autoavaliar e se aperfeiçoar, utilizando suas próprias respostas como fonte de sinal de recompensa.
No artigo Self-Rewarding Language Models, é proposto uma nova abordagem de alinhamento e auto-melhoria em que o próprio modelo atua como avaliador de suas respostas e gera seus próprios sinais de recompensa, eliminando a dependência de **anotações humanas** ou de **modelos de recompensa fixos**.

4. **Repositório, guias e tabelas comparativas**

A partir das novas leituras, atualizei todos os materiais que estão sendo produzidos.

Documentação completa

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Na próxima Semana, pretendo **iniciar a reprodução do estudo** *Self-Rewarding Language Models*, estruturando o ambiente de testes, selecionando o modelo base e implementando o pipeline de avaliação automática.

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Trilha RLAIF II - Prompt-Rewarding

1. Objetivo

Projetar funções de recompensa eficazes sempre foi um dos maiores desafios do aprendizado por reforço (RL). Métodos tradicionais exigem conhecimento especializado e ajustes manuais extensos, o que gera alto custo e risco de comportamentos não intencionais. Estudos recentes (Booth et al., 2023) indicam que mais de 90% dos profissionais ainda dependem de tentativa e erro no design de recompensas. Assim, o uso de LLMs como projetistas de recompensas busca automatizar e democratizar esse processo, reduzindo a dependência humana e ampliando a generalização.

Utilizar *Large Language Models* (LLMs) como avaliadores autônomos capazes de gerar, interpretar ou aplicar funções de recompensa durante a fase de alinhamento de modelos de linguagem, substituindo ou complementando o papel humano na atribuição de feedbacks. A ideia central é transformar o *prompting,* ou seja, a formulação de instruções textuais, em uma função de recompensa interpretável, que avalia as respostas do modelo com base em critérios explícitos, como utilidade, veracidade, clareza ou aderência ao objetivo.

2. Fichamento Resumo

RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback[34]

Em vez de treinar um modelo de recompensa (RM) a partir de pares de preferências que exigem anotações e re-treinamentos, o LLM no D-RLAIF atua diretamente como avaliador, atribuindo notas (1–10) às respostas geradas durante o RL. Essa pontuação é transformada em um valor escalar e usada como recompensa imediata no processo de otimização. Principais vantagens observadas:

- Resolve o problema de obsolescência do RM, pois o LLM avalia online cada nova resposta. Um RM treinado com dados de referência pode não ter visto alguns

[34] Idem 29

exemplos durante o alinhamento, o que pode fazer com que a recompensa seja enviesada.

- Elimina o ciclo demorado de rotulagem e re-treinamento de modelos de recompensa.

Eureka: Human-Level Reward Design via Coding Large Language Models[35]

O artigo propõe o *EUREKA*, um algoritmo universal de design de recompensas que utiliza LLMs para gerar funções de recompensa automáticas em tarefas de RL. O foco é alcançar desempenho humano na criação de recompensas, superando o processo manual e propenso a erros de engenharia tradicional. O EUREKA aproveita as capacidades de geração de código, raciocínio em contexto e melhoria iterativa dos LLMs para criar recompensas em formato de código executável. O LLM recebe como entrada o código do ambiente e a descrição da tarefa, e então gera funções de recompensa plausíveis *zero-shot*.

Text2Reward: Reward Shaping with Language Models for Reinforcement Learning[36]

O artigo propõe o *TEXT2REWARD*, um framework que usa LLMs para gerar automaticamente funções de recompensa densas e interpretáveis a partir de descrições em linguagem natural. A meta é substituir o design manual de recompensas, que exige expertise e é sujeito a erros, por um processo automático e generalizável.

Trilha RLAIF III - Self-Reward

1. Objetivo

Investigar e desenvolver métodos que permitam que modelos de linguagem aprendam a se autoavaliar e se aperfeiçoar, utilizando suas próprias respostas como fonte de sinal de recompensa. O propósito central é tornar os LLMs capazes de autonomia avaliativa e adaptativa, ou seja, que eles julguem a qualidade de suas próprias saídas, gerem critérios de recompensa sem depender de humanos e ajustem seus comportamentos com base

[35] MA, Yecheng Jason; et al. Eureka: Human-Level Reward Design via Coding Large Language Models. 2023. arXiv:2310.12931v2. Disponível em: https://arxiv.org/abs/2310.12931v2

[36] XIE, Tianbao et al. Text2Reward: Reward Shaping with Language Models for Reinforcement Learning. 2023. arXiv:2309.11489. Disponível em: https://arxiv.org/abs/2309.11489

nesses julgamentos. Esse paradigma busca superar as limitações do *Reinforcement Learning from Human Feedback (RLHF)* que depende de anotações humanas e avançar para uma forma de auto-alinhamento (*self-alignment*) e aprendizado contínuo, em que o próprio modelo se torna o agente que mede e impulsiona seu progresso.

2. Fichamento Resumo

Self-Rewarding Language Models[37]

O trabalho propõe uma nova abordagem de *alinhamento e auto-melhoria* para modelos de linguagem grandes (LLMs), em que o próprio modelo atua como avaliador de suas respostas e gera seus próprios sinais de recompensa, eliminando a dependência de anotações humanas ou de modelos de recompensa fixos. A meta é investigar se um LLM pode evoluir por conta própria, aprendendo simultaneamente a seguir instruções e a julgar a qualidade das respostas que produz, melhorando em ambos os aspectos a cada iteração

I. O modelo é treinado num ciclo iterativo chamado Iterative DPO (Direct Preference Optimization).
II. Em cada iteração:
    A. O modelo gera novas instruções e respostas (*self-instruct*).
    B. Ele mesmo atribui pontuações a essas respostas via *LLM-as-a-Judge*: um *prompt* que o faz agir como avaliador, justificando e dando nota (de 1 a 5) para cada resposta.
    C. As respostas com maior e menor pontuação formam pares de preferência usados para o próximo treinamento.
III. Assim, o modelo cria seu próprio dataset de preferências e treina sobre ele, aprimorando-se sem intervenção humana direta.

[37] Idem 8

APÊNDICE 4

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 15 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Utilizando RL para alinhamento de LLMs

Nas primeiras Semanas, me dediquei a compreender a área de Aprendizado por Reforço (RL) e iniciei uma aprofundamento em RL aplicado a LLMs, como resultado produzi materiais de apoio. Na Semana passada finalizei a trilha **Estudo sobre RLAIF** e como planejado, havia destacado o paper **Self-Rewarding Language Models** como principal objeto de estudo desta Semana.

1. **Planejamento:**

Comecei estabelecendo um planejamento mínimo para a reprodução. Os principais pontos que identifiquei:

   a. Preciso estudar o artigo a fundo,
   b. Refazer os experimentos (verificando se consigo replicá-lo **com exatidão**)
   c. Avaliar usando **as mesmas métricas do paper**, e
   d. Definir como eu poderia criticar, validar ou propor melhorias.

2. **Aprofundamento no artigo e nos materiais complementares**

Fiz uma releitura focada, verificando se o artigo estava bem estruturado e se tinha materiais complementares. Confirmei que os autores **não disponibilizaram código**. Por isso, diagramei o loop de treinamento proposto para entender melhor. LINK

Percebi que reproduzir integralmente seria **difícil**. Mas encontrei um repositório no GitHub da Oxen.ai - uma comunidade que leu o artigo e implementou o código. Agora meu foco mudou: executar o código deles, validar se está fiel ao paper e replicar os experimentos com as avaliações corretas. Disso, consegui fazer o seguinte:

   a. Dockerizar o repo para rodar na GPU da DGX - LINK
   b. Executando o primeiro script de treinamento com modelo menor (erros :'))

3. **Levantamento de hipóteses**

Com uma leitura mais crítica, identifiquei pontos que os autores não exploram a fundo. Levantei quatro

questões, mas quero focar em **pelo menos uma** para investigar:

Questão 1: Os dados de alinhamento limitaram os resultados?

- O artigo usa apenas 3.200 exemplos do Open Assistant, reconhecidamente fracos em raciocínio matemático
- **Por que é relevante:** Eles abrem o artigo falando sobre capacidade sobre-humana, mas concluem que o método não ajudou em raciocínio. Sem ablação dos componentes, não dá pra saber se os dados iniciais foram o gargalo.

Questão 2: DPO é limitante?

- **Por que é relevante:** DPO é comprovadamente bom para seguir instruções, mas pode não ser ideal para melhorar raciocínio. Sem ablação desse componente, fica a dúvida se outro algoritmo funcionaria melhor.

Questões 3: Quando essa abordagem para de funcionar? Por que pararam no M3?

Artigo treinou apenas até M3 e usou Llama 2 70B, isto é, cada modelo M1, M2 e M3 são uma versão melhorada do anterior.

Por que é relevante: Identificar quando/por que o método para de funcionar tem implicações práticas.

Questões 4: Esse método funciona para modelos menores?

Por que é relevante: O artigo usa Llama 2 70B, impraticável para a maioria dos pesquisadores.

Algumas observações:

- Percebi a impossibilidade de fazer uma reprodução exata, em vários aspectos, primeiro os dados de treinamento são uma amostra do Open Assistant, por fim, usar o Llama 2 70B mesmo quantizado por ser um desafio para quem tem hardware limitado.
- Todavia, o cerne da contribuição do artigo não está no modelo especificamente, mas no mecanismo de **self-rewarding iterativo.** Este mecanismo pode e deve ser testado em escalas acessíveis. Contudo, minhas hipóteses, por fim, estão limitadas e não conseguiria fazer uma correlação direta dos resultados da reprodução com o artigo.

DOCUMENTAÇÃO COMPLETA

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

- Continuar execução e verificação dos scripts de treinamento
- Acrescentar as avaliações no formato exato do paper (MT-Bench, reward modeling metrics)

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Reprodução Self-Rewarding

No artigo de YUAN et al.[38] é proposto que modelos de linguagem possam gerar seus próprios dados de treinamento e agir simultaneamente como avaliadores, superando a dependência de feedback humano típica do RLHF. A abordagem consiste em um ciclo iterativo em que um modelo inicial baseado em Llama-2-70B recebe um pequeno conjunto de dados seed, gera novos prompts, produz múltiplas respostas, avalia essas respostas via LLM-as-a-Judge e forma pares de preferência para treinar o próximo modelo por DPO. Em tese, cada iteração produz um modelo melhor em duas frentes: seguir instruções e atuar como avaliador, criando um “círculo virtuoso” de auto-aperfeiçoamento. O artigo apresenta uma premissa ambiciosa logo no resumo,a possibilidade de alcançar capacidades sobre-humanas através de auto-alinhamento iterativo. A analogia com AlphaZero no xadrez é tentadora, mas há uma diferença crucial: em xadrez temos métricas objetivas de vitória/derrota e um ambiente completamente observável. Em linguagem natural, não existe métrica objetiva universal de "qualidade". Algumas questões podem surgir naturalmente:

- Que forma teria um "conhecimento sobre-humano" em linguagem? Criaria novas estruturas linguísticas? Desenvolveria formas de raciocínio inatingíveis por humanos?
- Como validaríamos tal avanço? Os benchmarks atuais são projetados para avaliar a performance dentro do paradigma humano.

Nos resultados observamos que o modelo M3 melhorou em seguir instruções e gerar respostas mais elaboradas, mas não apresentou ganhos significativos em raciocínio. Nos benchmarks GSM8K e ARC-Challenge mantiveram-se similares ao baseline. O resultado indica que não houve degradação nessa capacidade, porém não apresenta nenhuma evidência sobre "conhecimento sobre-humano".

Durante a tentativa de reprodução, alguns problemas tornam-se mais evidentes. O artigo não fornece detalhes essenciais do pipeline indicando que a própria replicação depende de suposições e reconstruções. Por essa razão, embora tenha inicialmente

[38] Idem 8

planejado reproduzir o método, explorando ablações, verificando se funcionaria em modelos menores e investigando por que o processo parou no M3, concluí que a ausência de informações chave inviabiliza uma reprodução confiável. Diante disso, optei por pivotar para um segundo plano de experimentação com um baseline mais consistente e metodologicamente rastreável.

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 23 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Aprimorando de Modelos de Linguagem Grandes com Aprendizado por Reforço

Nas primeiras Semanas, me dediquei a compreender a área de Aprendizado por Reforço (RL) e iniciei uma aprofundamento em RL aplicado a LLMs, como resultado explorei como criar datasets usando outros LLMs, como recompensar LLMs no contexto de RL, datasets, benchmarks e ferramentas. Até então, meu foco vinha se estreitando para o uso de **LLMs auxiliares** no pipeline de alinhamento de modelos principais.

Na Semana passada, percebi a necessidade de retornar a uma base teórica sólida para estruturar as futuras experimentações. Retomei os artigos analisados desde a primeira semana e, após uma nova leitura, escolhi o *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning* como base metodológica principal. Essa escolha se deve ao fato de o artigo oferecer infraestrutura reprodutível, modelos abertos e lacunas explícitas para exploração por parte da comunidade.

Durante esta Semana, aprofundei a análise no artigo e identifiquei que os autores relatam que a **transferência de raciocínio de modelos grandes para modelos menores** (via destilação) é mais eficiente e econômica do que o pipeline original de RL, mas os modelos destilados receberam apenas **Supervised Fine-Tuning (SFT)**. Isso abre espaço para investigar o impacto de **RL pós-destilação**, o que será o foco das próximas etapas.

1. Baseline estabelecido

Foram utilizados os modelos de linguagem grandes destilados do DeepSeek-R1:

- **DeepSeek-R1-Distill-Llama-8B**
- **DeepSeek-R1-Distill-Qwen-7B**

O setup de avaliação foi construído com base no benchmark **AIME 2025**, aplicando o método **pass@k (k = 1 e 8)**, inspirado na metodologia do artigo original (AIME 2024).

Resultados:

- deepseek-ai/DeepSeek-R1-Distill-Llama-8B

- Pass@1: 20.00% (média de 1.00 tentativas) - 1h30 de execução
- Pass@8: 60.00% (média de 4.47 tentativas) - 8h

- deepseek-ai/DeepSeek-R1-Distill-Qwen-7B
  - Pass@1: 26.67% (média de 1.00 tentativas) - 30 min
  - Pass@8: 40.00% (média de 5.47 tentativas) - 2h30

2. Seleção de Datasets para Etapa de Alinhamento com RL

Defini dois datasets complementares para a próxima fase de experimentação:

1. ReClor — voltado à raciocínio lógico e compreensão de leitura, permitindo testar se o RL pós-destilação amplia a capacidade de inferência e coerência argumentativa.
2. Ultra Feedback — voltado a preferências e instruções humanas, útil para afinar o alinhamento instrucional e o comportamento do modelo em contextos abertos.

DOCUMENTAÇÃO COMPLETA

FORK REPOSITÓRIO OPEN-R1

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Nas próximas Semanas, pretendo dar continuidade às experimentações, com foco na implementação e avaliação do estágio de Aprendizado por Reforço nos modelos destilados.

1. Construção e teste de funções de recompensa
2. Adaptação do framework Open-R1 para treinar os modelos destilados utilizando os datasets previamente selecionados (*ReClor* e *UltraFeedback*), ajuste de parâmetros e integração o cálculo de recompensas personalizadas.
3. Avaliação dos checkpoints gerados, comparando o desempenho obtido após o treinamento por RL com o baseline estabelecido na etapa anterior

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Reprodução DeepSeek-R1

Diante das limitações encontradas na tentativa de reproduzir trabalhos anteriores optei por adotar o DeepSeek-R1[39] como referência principal para as minhas experimentações. Este trabalho oferece infraestrutura reprodutível, código aberto e checkpoints oficiais, reduzindo incertezas metodológicas e permitindo construir uma linha de experimentação sólida. Além disso, os próprios autores identificam lacunas abertas à investigação, em especial o fato de que os modelos destilados receberam apenas *SFT*, deixando explicitamente em aberto a etapa de *RL* como oportunidade para a comunidade. O artigo descreve três abordagens distintas:

1. DeepSeek-R1-Zero: RL aplicado diretamente ao modelo base, sem SFT prévio.
2. DeepSeek-R1: RL aplicado após um extenso ajuste via Cadeia de Pensamento (CoT).
3. Destilação de Capacidade de Raciocínio: transferência do raciocínio do modelo grande para modelos menores e densos.

Entre essas opções, selecionei a Abordagem 3 (Destilação) por equilibrar viabilidade computacional, relevância científica e por alinhar-se aos eixos centrais do meu estudo: RLAIF, destilação de raciocínio e alinhamento de LLMs. A ausência de RL no pipeline de destilação constitui, portanto, uma oportunidade direta de contribuição experimental.
Com base nisso, minha hipótese de investigação passa a ser:

- O Aprendizado por Reforço aplicado após a destilação pode ampliar as capacidades de raciocínio lógico dos modelos?
- O Aprendizado por Reforço pode ampliar a capacidade do modelo de alinhar-se às preferências humanas?

Planejamentos da Experimentação

Durante a Semana 8, também desenvolvi um plano de experimentação e avalie os modelos bases escolhidos.

Fase 1 - Baseline

[39] Idem 11

1. deepseek-ai/DeepSeek-R1-Distill-Llama-8B[40]
    a. Pass@1: 20,00% (1h30)
    b. Pass@8: 60,00% (8h)
2. deepseek-ai/DeepSeek-R1-Distill-Qwen-7B[41]
    a. Pass@1: 26,67% (30 min)
    b. Pass@8: 40,00% (2h30)

Obs: amostragem com pass@k (k = 1 e k = 8) com o benchmark OpenCompass/AIME2025[42] (primeiras 15 questões matemáticas).

Fase 2 - Seleção de Datasets para RL

Para a próxima etapa de alinhamento por RL, selecionei o ReClor, focado em raciocínio lógico e compreensão de leitura, adequado para testar ganhos em inferência e coerência como dataset para treinamento.

Fase 3 - Alinhamento por RL

Construção das funções de recompensas necessárias, adaptação do framework Open-R1[43]e execução do treinamento

Fase 4 - Avaliação

Avaliação do modelo treinado e comparação com a baseline da Fase 1.

[40] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B
[41] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-7B
[42] https://huggingface.co/datasets/opencompass/AIME2025
[43] https://github.com/huggingface/open-r1

APÊNDICE 5

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 6 de nov. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Aprimorando de Modelos de Linguagem Grandes com Aprendizado por Reforço

Nas Semanas anteriores, realizei um estudo sobre aprimoramento de modelos de linguagem grandes com o uso de Aprendizado por Reforço. Esse aprimoramento geralmente visa melhorar algum elemento do modelo, seja produzir respostas mais adequadas ao "gosto" humano ou aperfeiçoar habilidades específicas como: raciocínio, chamamento de ferramentas, entre outras.

- Ao ler o artigo do DeepSeek-R1, identifiquei uma oportunidade: os autores sugerem que seus modelos destilados (treinados apenas com SFT) poderiam melhorar com uma etapa adicional de RL.
- Planejei experimentações com dois focos: **raciocínio** e **preferências humanas.** Devido ao custo computacional, escolhi trabalhar apenas com o DeepSeek-R1-Distill-Llama-8B, que apresentou melhor desempenho inicial (pass@k).
- Nesta semana, executei o primeiro experimento focado em raciocínio, buscando melhorar o desempenho no AIME 2025, especialmente na métrica pass@1 (baseline de apenas 20%).

Fase 1 - Baseline

Fase 2 - Dataset + Recompensas
Utilizei o dataset ReClor (contexto, perguntas e alternativas) e aproveitei as funções de recompensa do repositório Open-R1.

Fase 3 - Treinamento
Algumas adaptações no repositório (Dockerfile, função que recebe o dataset formatado e algumas funções de recompensa teste), ajustes de hiperparâmetros, como no caso da quantidade de gerações por steps que depende fortemente da disponibilidade de GPUS.

Fase 4 - Avaliação

- Utilizei os gráficos de média e desvio padrão das recompensas para fazer uma análise prévia dos resultados, durante o treinamento e antes da avaliação com o benchmark estabelecido.
- Diagnostique que três funções de recompensa estão "estagnadas" e vou aprofundar no entendimento delas durante a próxima semana, pretendo investigar se foi algum erro de implementação ou se existe um limite de funções de recompensas ideal, e se quando excedemos o

modelo foca em recompensas mais “recompensadoras”.

- Resumindo o comportamento observado no treinamento: não espero uma mudança muito significativa no benchmark. Isso porque perdi um treinamento que havia chegado a 380 steps (onde os gráficos começavam a mostrar sinais de melhoria e estabilização) e, ao reiniciar, reduzi a quantidade para apenas 250 steps por restrições recursos computacionais. Observando os gráficos dessa nova execução, ficou evidente que o modelo precisava de mais steps para convergir adequadamente.
- **deepseek-ai/DeepSeek-R1-Distill-Llama-8B**
  - Pass@1: 20.00% - 3 questões certas (média de 1.00 tentativas) - 1h30
  - Pass@8: 60.00% - 9 questões certas (média de 4.47 tentativas) - 8h
- **Llama-3.1-8B-Residencia-1k (mil amostras)**
  - Pass@1: 26.67% - 4 questões certas (média de 1.00 tentativas) - 2h
  - Pass@8: 46.67% - 7 questões certas (média de 5.07 tentativas) - 5h

No baseline, 20% representa 3 questões corretas e o modelo com RL acertou 4 questões. Esse resultado é positivo, demonstrando que a capacidade inicial de raciocínio do modelo teve uma melhora. Já no caso do pass@8, o resultado foi preocupante, o modelo com RL precisou de mais tentativas para acertar as 7 questões, enquanto que o modelo base acertou 9 questões com 4.47 tentativas em média.

Algumas observações sobre esse resultado: o dataset AIME usado para avaliação é pequeno e uma questão a mais ou a menos representa apenas 6.7% de diferença; o modelo de RL também pode ter se tornado mais “rígido”, admitindo pouca exploração, algo a se verificar. Ou ainda, o excesso de funções de recompensa e a quantidade de steps podem ter provocado esse comportamento, fortalecendo demais um tipo de raciocínio.

DeepSeek-R1 Llama 3.1 Distill 8B + Aprendizado por reforço

DOCUMENTAÇÃO COMPLETA

FORK OPEN-R1

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Na próxima Semana, pretendo executar novamente a experimentação, porém com um objetivo distinto: **ajustar o LLM para melhor responder às preferências humanas**. Algumas questões levantadas

1. Como essa etapa adicional de alinhamento impactará o desempenho do modelo?
2. Podemos esperar uma melhoria nos acertos do AIME 2025 ou o desempenho se manterá no mesmo patamar?
3. É possível que haja até uma degradação em tarefas específicas de raciocínio matemático?

Utilizarei o mesmo modelo para esse treinamento e manterei também o mesmo benchmark garantindo que os resultados possam ser comparados com os anteriores. Para o treinamento, usarei uma amostra do dataset UltraFeedback e proporei novas funções de recompensa que capturem aspectos de utilidade, clareza, segurança e alinhamento com a expectativa do usuário.

Observação: [caso precise fazer alguma observação, de qualquer “natureza”]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Reasoning

- Fase 1 - Baseline

A primeira etapa consistiu na definição de uma linha de base utilizando o modelo deepseek-ai/DeepSeek-R1-Distill-Llama-8B[44]. O benchmark inicial foi conduzido no OpenCompass/AIME2025[45], avaliando as 15 primeiras questões matemáticas do dataset. O desempenho foi medido por meio do script *pass_at_k*, configurado com k = 1 e k = 8, permitindo observar tanto a capacidade determinística do modelo quanto sua performance sob amostragem ampliada. Esse baseline serviu como referência para as fases posteriores de alinhamento por aprendizagem por reforço (RL).

- Fase 2 - Preparação para RL

Na preparação para o treinamento por RL, selecionei o dataset ReClor[46], apropriado para tarefas de raciocínio lógico. A construção do conjunto de recompensas foi intensamente baseada nas implementações disponíveis no repositório Open-R1[47], cuja finalidade é justamente reproduzir o comportamento do DeepSeek-R1. As recompensas utilizadas foram:

- accuracy_reasoning: avalia se a conclusão do modelo coincide com a resposta correta de referência.
- reasoning_steps_reward: verifica a presença de um raciocínio estruturado, claro e passo a passo.
- len_reward: penaliza respostas excessivamente longas para conter a tendência dos LLMs treinados com RL de inflar o raciocínio artificialmente.
- tag_count_reward: verifica a conformidade com a formatação esperada pelo format_reward(), considerando as tags think e answer.

Esse conjunto buscou equilibrar qualidade, estruturação e eficiência das respostas, embora já fosse esperado que múltiplas recompensas simultâneas pudessem gerar conflitos de sinal e instabilidade durante o treinamento.

[44] Idem 40
[45] Idem 42
[46] Idem 43
[47] YU, Weihao et al. ReClor: A Reading Comprehension Dataset Requiring Logical Reasoning. ICLR, 2020. Disponível em: https://openreview.net/pdf?id=HJgJtT4tvB

- Fase 3 — Alinhamento por RL

O alinhamento foi realizado adaptando o framework Open-R1 para treinar o modelo destilado utilizando GRPO (Generative Reinforcement Policy Optimization). As principais modificações envolveram:

1. Ajustes na receita de treinamento, incluindo hiperparâmetros específicos ao modelo e às restrições computacionais disponíveis.
2. Criação de um Dockerfile customizado, garantindo reprodutibilidade do ambiente, isolamento de dependências e execução consistente em diferentes infraestruturas.
3. Adaptação do dataset ReClor[48] ao pipeline do GRPO, ajustando scripts de pré-processamento e mapeamento de contexto, questões e alternativas.

A primeira execução alcançou apenas 379 de 1160 steps (≈33%) antes de ser interrompida, devido a uma configuração inadequada de checkpoints — que estavam sendo salvos por época, e não por step. O treino levou 10h29min até a interrupção, com previsão de mais ~20h para finalizar. Após essa falha, corrigi o mecanismo de salvamento e reduzi o dataset para um subset menor, permitindo treinos mais curtos e menos suscetíveis à perda de progresso.

- Fase 4 — Avaliação

A avaliação envolveu análise detalhada dos resultados, comparando os checkpoints obtidos com o baseline da Fase 1. Foram identificados padrões importantes no comportamento das recompensas:

Análises Gráficas

Os gráficos foram observados, mas a análise só foi realizada após a finalização dos steps. Neste primeiro caso, no entanto, tivemos o treinamento interrompido o que culminou em 379° steps de treinamentos realizados.

[48] Dataset

Tag Count Reward

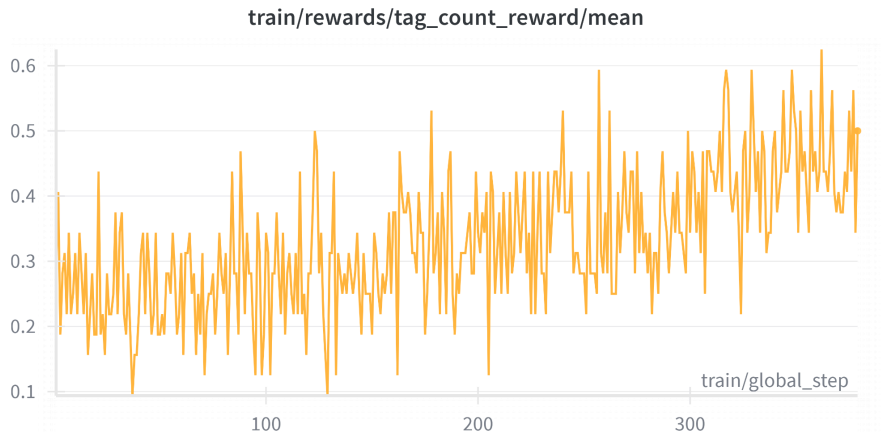


Fig 1: Média da função de recompensa que verifica se produzimos o número desejado de tags think e answer associadas a format_reward().

Nos primeiros steps, a métrica apresentou alta instabilidade, o que é esperado enquanto o modelo aprende a estrutura exigida. A partir do step 210, observou-se melhora gradual, com convergência parcial entre 0.2 e 0.6. Entre os steps 350 e 379, o desempenho continuava melhorando, indicando que o modelo estava internalizando o uso das tags. Ajustes no chat template podem potencializar esse aprendizado.

Len Reward

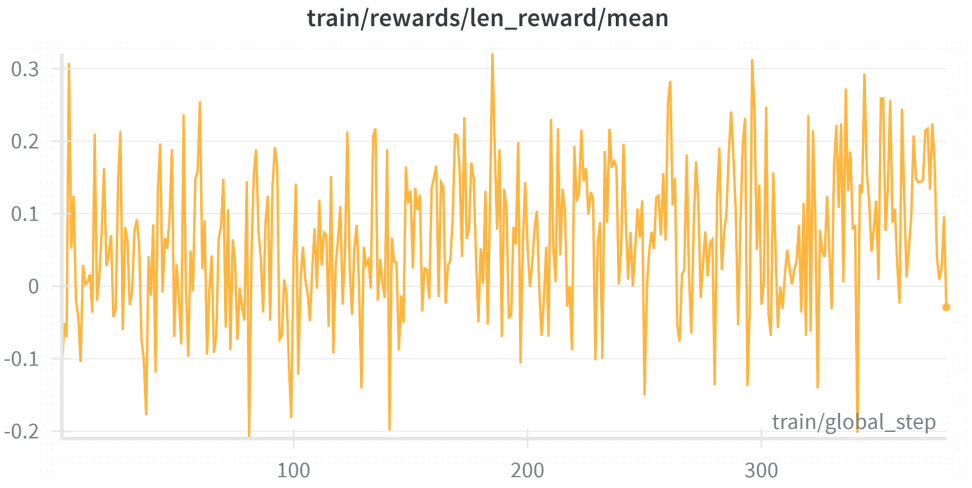


Fig. 2: Média da função que calcula recompensas baseadas no comprimento para desencorajar o excesso de reflexão e promover a eficiência dos tokens.

A recompensa de comprimento oscilou fortemente entre valores positivos e negativos, sem tendência de estabilização. Isso sugere possíveis conflitos com outras recompensas ou sinal insuficiente para guiar o comportamento do modelo. É provável que a penalização atual esteja subótima ou que sua interação com as demais esteja obscurecendo seu efeito.

Reasoning Steps Reward

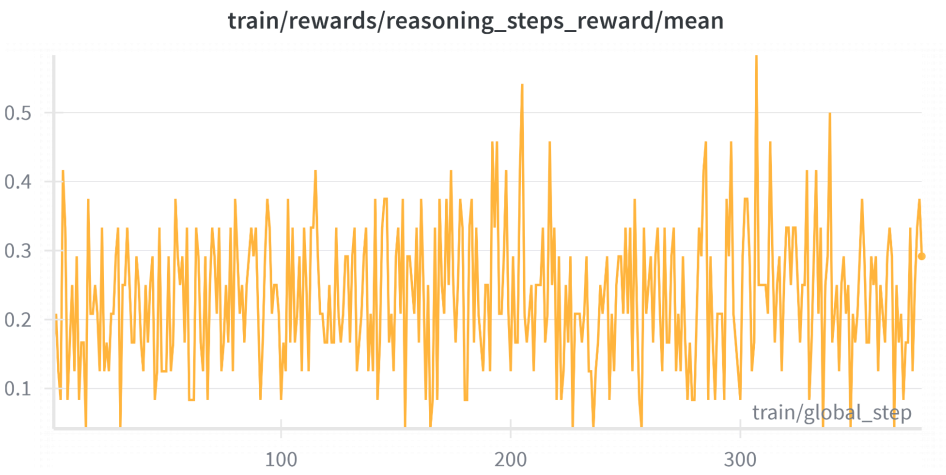
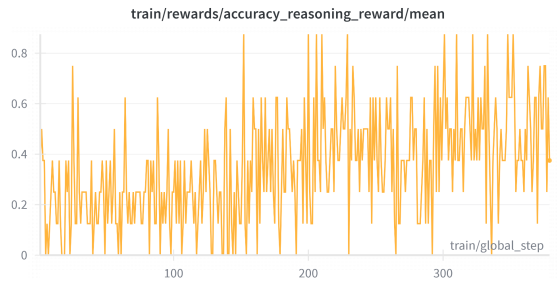


Fig. 3: Média da função de recompensa que verifica o raciocínio claro passo a passo.

Não houve evidência de melhoria consistente. O comportamento indica que o modelo não convergiu para um padrão de raciocínio mais estruturado, reforçando a necessidade de revisar o balanceamento entre recompensas ou mesmo reduzir sua quantidade.

Accuracy Reasoning Reward



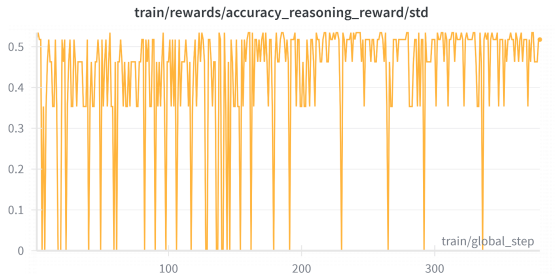


Fig. 4 e 5: Média e desvio padrão da função de recompensa que verifica se a conclusão é a mesma que a verdade de referência.

No início, a acurácia variava entre 0 e 0.8, com quedas abruptas e desvio padrão instável. Entretanto, nos últimos steps do treino interrompido, houve aumento de estabilidade: a média manteve-se mais próxima de valores intermediários e o desvio padrão convergiu para a faixa de 0.4 a 0.5. Esse comportamento sugere um progresso inicial do modelo em direção à precisão, embora ainda limitado pelo curto tempo de treino.

Análises Gráficas II

Uma segunda análise foi conduzida a partir de um treinamento com 250 steps e subset de 1.000 amostras, totalizando cerca de 7 horas. Os resultados foram comparados à execução anterior:

- Tag Count Reward: apresentou comportamento semelhante, com instabilidade inicial e estabilização parcial.
- Accuracy Reasoning Reward: novamente exibiu grande volatilidade antes do step 300, o que explica a performance modesta do modelo V1. Essa instabilidade inicial indica que a tarefa apresenta complexidade significativa para o modelo em estágio inicial de RL.

Ambas análises apontam para um padrão consistente: as recompensas apresentam aprendizado parcial, mas insuficiente para consolidar melhorias significativas no raciocínio antes dos steps em que o treinamento foi interrompido.

Tag Count Reward

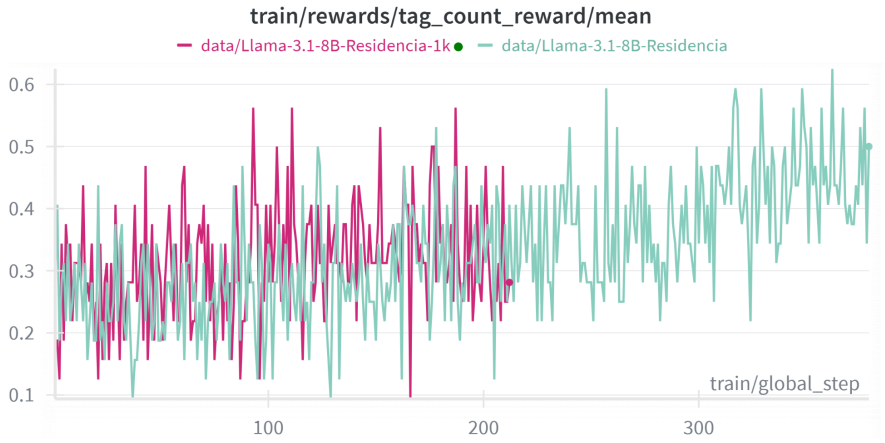


Fig. 6: A recompensa apresentada em rosa representa o treinamento com mil amostras antes de completar as 250 amostras. Porém, percebemos a similaridade entre os dois resultados.

Accuracy Reasoning Reward

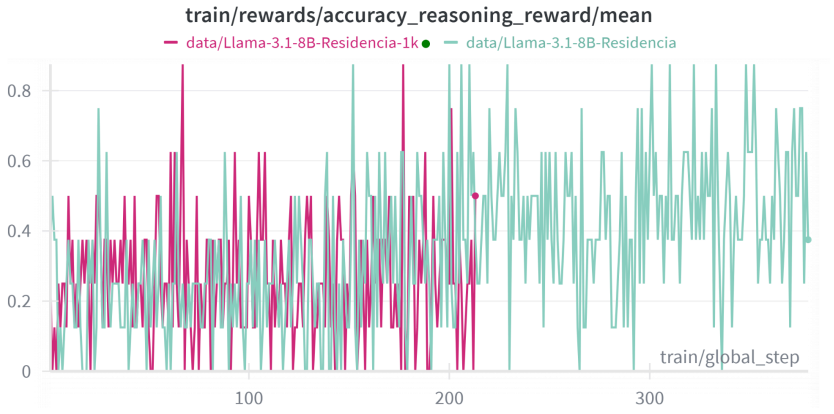


Fig. 7: Média da função de acurácia que na cor rosa representa o treinamento com 1k amostras.

Resultados Reasoning

- deepseek-ai/DeepSeek-R1-Distill-Llama-8B
  - Pass@1: 20.00% (média de 1.00 tentativas) - 1h30
  - Pass@8: 60.00% (média de 4.47 tentativas) - 8h
- LisandraMoura/Llama-3.1-8B-Residencia-1k
  - Pass@1: 26.67% (média de 1.00 tentativas) - 2h

- Pass@8: 46.67% (média de 5.07 tentativas) - 5h

Os resultados evidenciam que, embora haja sinais de aprendizado nas recompensas, a performance em raciocínio matemático permanece limitada, possivelmente devido ao tamanho reduzido do treino, interrupções constantes e instabilidade das métricas.

Reflexões e Próximos Passos

Os experimentos revelam dois fatores críticos:

1. A dificuldade de projetar recompensas robustas e livres de ambiguidades;
2. A influência direta do tempo de treinamento e disponibilidade de hardware no progresso do modelo.

Para a próxima semana, pretendo repetir o ciclo experimental, agora com foco em ajustar o modelo para preferências humanas. Isso levanta questões importantes:

1. Como o alinhamento por preferências afetará o desempenho no AIME 2025?
2. As respostas tenderão a se tornar mais consistentes, ou haverá degradação em tarefas especializadas de raciocínio matemático?
3. Até que ponto o trade-off entre alignment e reasoning será perceptível?

Essas questões orientaram a próxima fase do trabalho, buscando compreender os impactos do alinhamento humano no desempenho de LLMs em tarefas de alta complexidade.

APÊNDICE 6

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 12 de nov. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

Lisandra Cristina de Moura Menezes

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Aprimorando de Modelos de Linguagem Grandes com Aprendizado por Reforço

Nas Semanas anteriores, realizei um estudo sobre aprimoramento de modelos de linguagem grandes com o uso de Aprendizado por Reforço. Esse aprimoramento geralmente visa melhorar algum elemento do modelo, seja produzir respostas mais adequadas ao "gosto" humano ou aperfeiçoar habilidades específicas como: raciocínio, chamamento de ferramentas, entre outras.

- Ao ler o artigo do DeepSeek-R1, identifiquei uma oportunidade: os autores sugerem que seus modelos destilados (treinados apenas com SFT) poderiam melhorar com uma etapa adicional de RL.
- Planejei experimentações com dois focos: **raciocínio S9** e **preferências humanas S10.** Devido ao custo computacional, escolhi trabalhar apenas com o DeepSeek-R1-Distill-Llama-8B, que apresentou melhor desempenho inicial (pass@k).

Fase 1 - Baseline S8
Avaliei o modelo base `deepseek-ai/DeepSeek-R1-Distill-Llama-8B` com o benchmark o *OpenCompass/AIME2025*, focando nas primeiras 15 questões matemáticas.
A avaliação foi conduzida através do script `pass_at_k`, testando diferentes amostragens.

Fase 2 - Dataset + Recompensas
S9 - Utilizei o dataset ReClor (contexto, perguntas e alternativas) e aproveitei as funções de recompensa do repositório Open-R1.
S10 - Optei por preservar a mesma base experimental utilizada na S9, assegurando consistência metodológica e comparabilidade direta entre os resultados, mantendo o ReClor como dataset central, em vez de migrar para o *UltraFeedback*, uma vez que este último não facilitaria a implementação da avaliação de preferência, então preferi seguir usando o dataset de reasoning e encontrar uma boa forma de recompensar próximo do gosto humano. Assim, mantive as funções de recompensa de raciocínio e adicionei as novas funções de avaliação propostas nesta etapa.

1. **accuracy_reasoning**: Função de recompensa que verifica se a conclusão é a mesma que a verdade de referência.
2. **Reasoning_steps_reward:** Função de recompensa que verifica o raciocínio claro passo a passo.
3. **Tag_count_reward:** Função de recompensa que verifica se produzimos o número desejado de

tags think e answer associadas a `format_reward()`.

4. **Len_reward**: Calcule recompensas baseadas no comprimento para desencorajar o excesso de reflexão e promover a eficiência dos tokens. [*REMOVIDO NA S10]*
5. **Openai_judge:** avalia dimensões humanas de qualidade via LLM-as-a-Judge.

OBS: Na Semana que se decorreu, testei **três abordagens de função de recompensa** para promover o **alinhamento com preferências humanas**. Primeiro, ao analisar o dataset, percebi que ele foi projetado para o **DPO**, que usa comparações entre respostas escolhidas e rejeitadas, enquanto o **GRPO**, utilizado no meu experimento, exige **valores absolutos de recompensa**. A princípio, tentei encontrar uma forma de utilizar os dados do **UltraFeedback**, mas logo percebi que não era uma tarefa trivial atribuir um valor absoluto a aspectos como **utilidade** e **honestidade**, entre outros. Diante disso, retornei às minhas tabelas comparativas em busca de possíveis soluções.

1. Tentei inicialmente modelar recompensas para capturar utilidade, avaliando se a resposta apresentava pensamento passo a passo, tinha extensão adequada e oferecia explicações suficientes para ser de fato útil. Porém, essa estratégia rapidamente se mostrou limitada e difícil de validar rigorosamente.
2. Em seguida, busquei empregar um **modelo de recompensa treinado no UltraFeedback**, mas enfrentei **limitações de memória**.
3. Por fim, adotei a estratégia de **LLM-as-a-Judge**, utilizando um **modelo via API** capaz de avaliar utilidade e qualidade de forma semelhante a um humano. D-RLAIF - Prompt Reward

Fase 3 - Treinamento
Utilizei o framework Open-R1 com GRPO, com configuração similar à semana anterior, tendo como diferencial principal a adição da reward function openai_judge.

1. Número de Gerações : 2
2. Quantidade de Steps: 250
3. Dataset ReClor com 1k amostras

Fase 4 - Avaliação
Monitorei durante o treinamento gráficos de média e desvio padrão das recompensas para fazer uma análise prévia dos resultados, durante o treinamento e antes da avaliação com o benchmark estabelecido. Link

- **deepseek-ai/DeepSeek-R1-Distill-Llama-8B**
  - Pass@1: 20.00% - 3 questões certas (média de 1.00 tentativas) - 1h30
  - Pass@8: 60.00% - 9 questões certas (média de 4.47 tentativas) - 8h
- **Llama-3.1-8B-Residencia-1k (mil amostras)**
  - Pass@1: 26.67% - 4 questões certas (média de 1.00 tentativas) - 2h
  - Pass@8: 46.67% - 7 questões certas (média de 5.07 tentativas) - 5h
- **Llama-3.1-8B-Residencia-1k-V2 (mil amostras)**
  - **Pass@1: 40.00% (média de 1.00 tentativas) - 1h30**
  - **Pass@8: 53.33% (média de 4.60 tentativas) - 8h30**

DOCUMENTAÇÃO COMPLETA

FORK OPEN-R1

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

Dado a finalização das stages e gates, pretendo agora compilar todo meu trabalho no tcc.

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

Do backlog, ficam alguns aprendizados obtidos nesta fase de experimentação, que evidenciaram lacunas na minha trilha inicial de estudos, especialmente no que diz respeito a técnicas para modelar helpfulness, honesty e instruction-following de forma mais robusta e mensurável, dado que preferências humanas são fundamentalmente mais difíceis de modelar que reasoning.

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Preferências Humanas

Dando continuidade ao planejamento da semana anterior, após concluir o primeiro treinamento focado em raciocínio lógico com funções de recompensa voltadas ao pensamento passo a passo, o objetivo desta etapa foi investigar um segundo eixo relevante no RL aplicado a LLMs: o alinhamento às preferências humanas.

Semanas antes, eu havia selecionado o UltraFeedback, um dataset composto por instruções, múltiplas respostas de diferentes LLMs e anotações numéricas geradas por avaliadores automáticos. Cada instância contém uma instrução, diversas respostas e métricas de helpfulness, honesty e instruction-following.

Identifiquei rapidamente um desafio estrutural: o UltraFeedback é otimizado para métodos como DPO (Direct Preference Optimization), que dependem de comparações relativas entre respostas (melhor vs. pior). Já algoritmos como GRPO/PPO exigem recompensas absolutas, isto é, valores numéricos contínuos. Não bastava saber que "resposta A é melhor que B"; era necessário traduzir preferências relativas em scores adequados para otimização.

Minha primeira tentativa consistiu em projetar heurísticas próprias para avaliar helpfulness. Considerei critérios como presença de raciocínio estruturado, comprimento apropriado da resposta e correção factual. Embora funcionais, essas heurísticas eram limitadas: não capturavam aspectos subjetivos, exigiam calibração extensiva e não refletiam adequadamente dimensões como honestidade ou segurança.

Ao revisar minha tabela comparativa de modelos de recompensa, encontrei o UltraCM-13B, treinado especificamente sobre o UltraFeedback para gerar notas contínuas nas três dimensões principais. Teoricamente, seria a solução ideal. Porém, mesmo utilizando sua versão reduzida, o modelo apresentou erros frequentes de Out of Memory, especialmente durante picos de alocação, tornando inviável sua integração ao pipeline de RL.

Diante desses limites, adotei a estratégia de Prompt Rewarding, alinhada à literatura sob o paradigma LLM-as-a-Judge. Trabalhos recentes mostram alta correlação entre julgamentos de LLMs e avaliações humanas. Essa abordagem permitiu criar recompensas por meio de API, evitando sobrecarga na GPU. O avaliador é instruído via prompt

especializado a julgar dimensões como utilidade, clareza, aderência a instruções e segurança.

Fase 1 - Baseline

Mantive o mesmo baseline experimental da semana anterior: o modelo deepseek-ai/DeepSeek-R1-Distill-Llama-8B, avaliado no OpenCompass/AIME2025 utilizando o script pass_at_k com k = 1 e k = 8.

Fase 2 - Preparação para RL

Apesar da introdução de objetivos relacionados a preferências humanas, mantive o ReClor como dataset central, garantindo consistência metodológica com o experimento da semana anterior. Assim, foi possível comparar diretamente:

- Semana 9 com o objetivo de RL para reasoning
- Semana 10 com o objetivo de RL para preferências humanas

As recompensas utilizadas foram:

- accuracy_reasoning - confere se a conclusão coincide com a resposta correta.
- reasoning_steps_reward - verifica se há raciocínio passo a passo.
- tag_count_reward - avalia conformidade com o formato think/answer.
- openai_judge - responsável pela avaliação holística de qualidade via LLM-julgador.

Essa decisão priorizou coerência experimental e permitiu análise controlada dos efeitos introduzidos pela dimensão humana do alinhamento.

Fase 3 - Alinhamento por RL

Utilizei o framework Open-R1 com GRPO, com configuração similar à semana anterior, tendo como diferencial principal a adição da reward function openai_judge.

Fase 4 - Avaliação

A avaliação envolveu análise detalhada dos resultados, comparando os checkpoints obtidos com o baseline da Fase 1.

Análises Gráficas

Openai Judge Reward

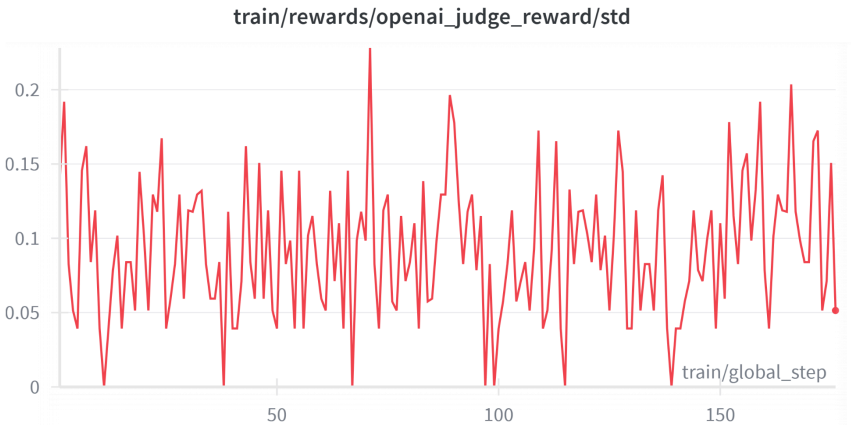


Fig. 8: Desvio Padrão da nova recompensa implementada e que Avalia dimensões humanas de qualidade via LLM-as-a-Judge.

A variabilidade permaneceu baixa (≈ 0.05–0.15), indicando consistência na avaliação humana simulada ao longo do processo.

KL-divergence

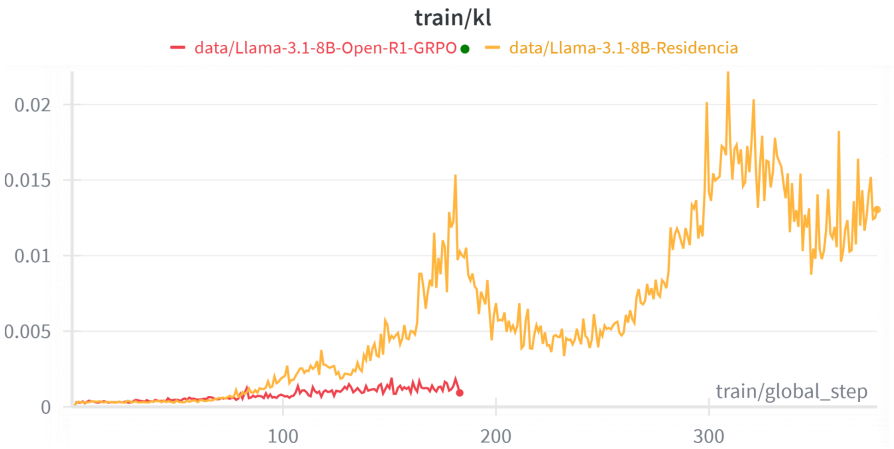


Fig. 9: A divergência KL, também chamada de entropia relativa, é uma medida estatística que quantifica o quanto uma distribuição de probabilidade Q difere de uma distribuição verdadeira P.

A KL se manteve baixa e estável no modelo V2, ao contrário do V1, que apresentou aumento acentuado. Isso sugere alinhamento mais natural, com menor risco de *reward hacking* e menor desvio da distribuição original.

Accuracy Reasoning Reward

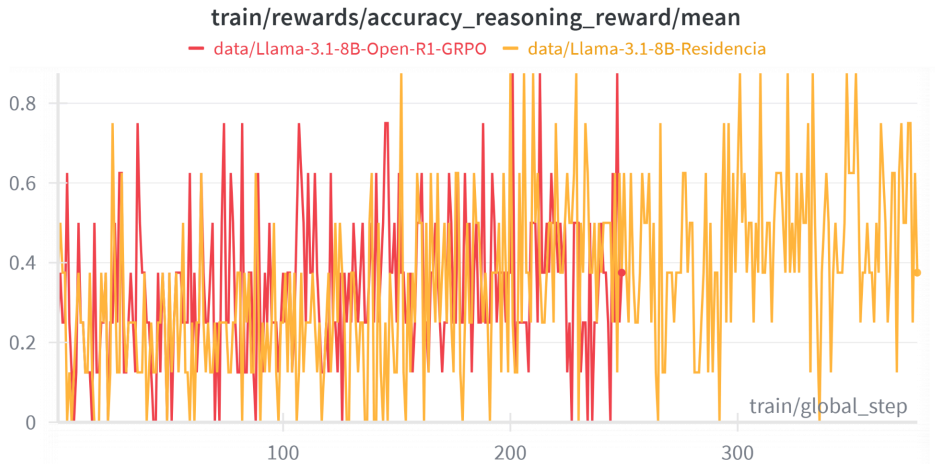


Fig. 10: Média da função de recompensa que verifica se a conclusão é a mesma que a verdade de referência.

A curta duração do treino impediu a estabilização; as flutuações permaneceram elevadas. Execuções mais longas tendem a mostrar convergência mais clara.

Tag Count Reward

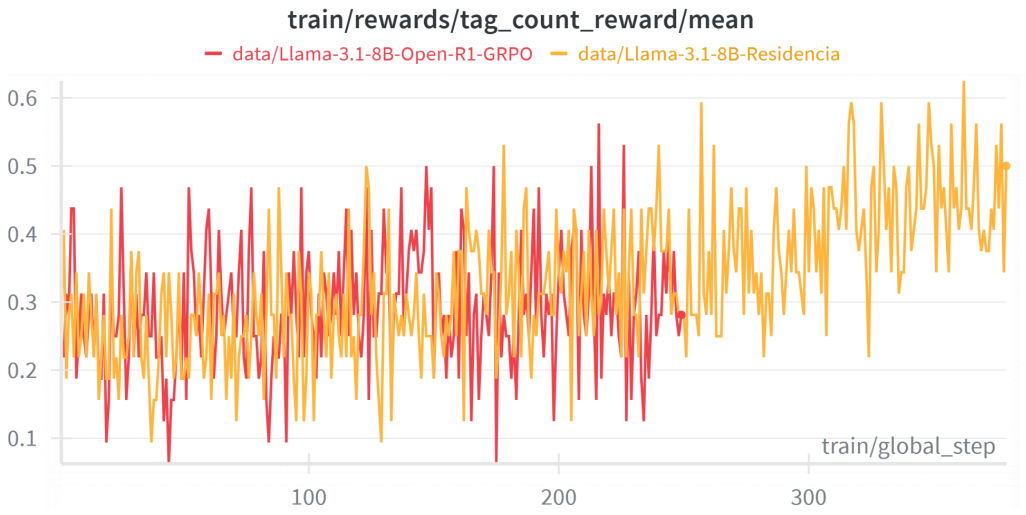


Fig. 11: Função de recompensa que verifica se produzimos o número desejado de tags think e answer associadas a format_reward().

As recompensas permaneceram estáveis entre 0.3 e 0.4, indicando rápida internalização do formato *think/answer*.

Reasoning Steps Reward

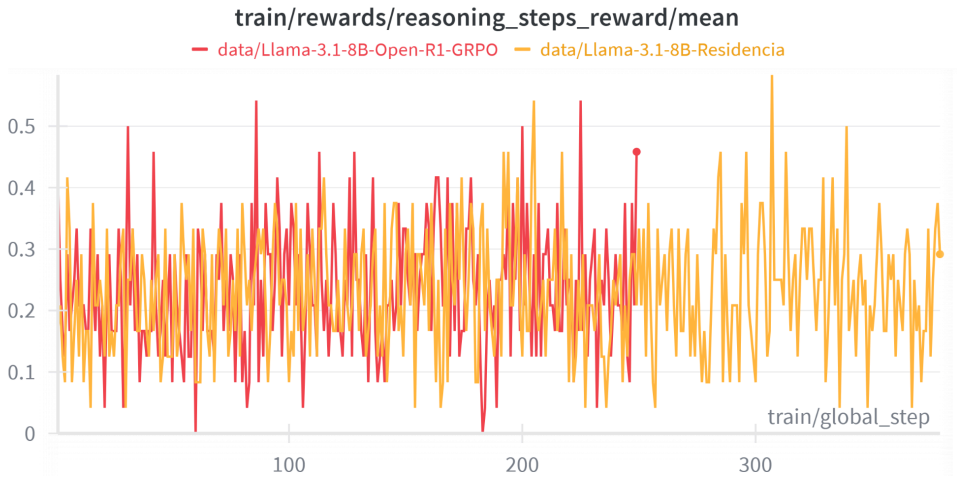


Fig. 12: Função de recompensa que verifica o raciocínio claro passo a passo.

A métrica permaneceu estável, porém sem tendência ascendente, provavelmente devido ao número reduzido de steps.

Resultados Preferências Humanas

1. deepseek-ai/DeepSeek-R1-Distill-Llama-8B
    - Pass@1: 20.00% (média de 1.00 tentativas) - 1h30
    - Pass@8: 60.00% (média de 4.47 tentativas) - 8h
- LisandraMoura/Llama-3.1-8B-Residencia-1k (V1)
    - Pass@1: 26.67% (média de 1.00 tentativas) - 2h
    - Pass@8: 46.67% (média de 5.07 tentativas) - 5h

1. Llama-3.1-8B-Residencia-1k-V2 (V2)
    1. Pass@1: 40.00% (média de 1.00 tentativas) - 1h30
    2. Pass@8: 53.33% (média de 4.60 tentativas) - 8h30

O modelo V2 apresentou o maior ganho absoluto de toda a experimentação: +13.33 pontos percentuais em relação ao V1 e +20 pontos em relação ao baseline. Esse resultado contraria hipóteses céticas sobre potenciais vieses do LLM-as-a-Judge. Pelo contrário, sugere que uma avaliação holística conseguiu captar aspectos qualitativos que as recompensas específicas (como reasoning_steps_reward ou len_reward) não modelaram adequadamente. A dimensão de clareza do raciocínio, fortemente enfatizada no prompt do judge, parece ter incentivado o modelo a estruturar respostas mais coerentes, contribuindo para a melhoria observada.

Um padrão particularmente relevante surge ao comparar Pass@1 e Pass@8:

- A maior melhoria ocorreu em Pass@1.
- O Pass@8 permaneceu dentro do intervalo típico (46–60%).

Esse comportamento indica que o modelo aprendeu a produzir a resposta correta de forma mais consistente na primeira tentativa, e não apenas a aumentar a probabilidade de acerto ao longo de diversas amostras. Trata-se de um sinal forte de aprendizagem estratégica, e não de exploração estocástica do espaço de respostas.

Com o dataset já organizado[49] e o repositório[50] devidamente estruturado, qualquer pessoa pode reproduzir exatamente os mesmos experimentos conduzidos na reprodução. Além disso, é possível testar os modelos V1[51] e V2[52] diretamente a partir dos pesos disponibilizados no Hugging Face.

[49] https://huggingface.co/datasets/LisandraMoura/reclor-sample-1k
[50] https://github.com/LisandraMoura/open-r1
[51] https://huggingface.co/LisandraMoura/Llama-3.1-8B-Residencia
[52] https://huggingface.co/LisandraMoura/Llama-3.1-8B-Residencia-1k-Merged