Self-Rewarding Language Models

Weizhe Yuan[1,2] Richard Yuanzhe Pang[1,2] Kyunghyun Cho[2]
Xian Li[1] Sainbayar Sukhbaatar[1] Jing Xu[1] Jason Weston[1,2]

[1] Meta [2] NYU

ICML 2024

이정섭

KOREA UNIVERSITY

Background

1. Alignment Tax

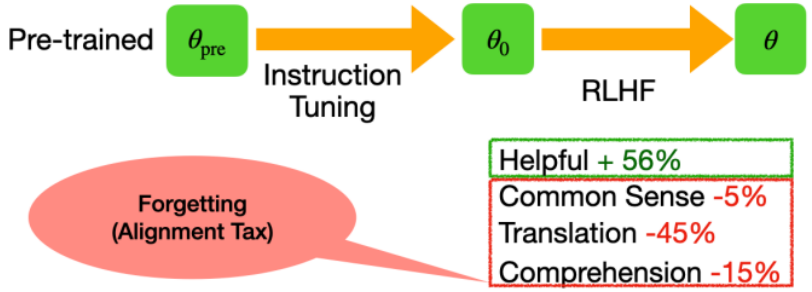


Figure 1: Illustration of RLHF procedure and the alignment tax.

- Training language models to follow instructions with human feedback (NIPS 2022, InstructGPT)
  ➔ RLHF 제안, alignment tax 정의
- Mitigating the Alignment Tax of RLHF (arXiv 2024)
  ➔ RLHF에서 발생하는 Alignment Tax 완화 연구

Introduction

초인적인 LLM 모델 학습을 위해 super human feedback이 필요

➔ 현재의 접근 방법은 주로 **인간의 선호도(human preference data)로 보상 모델을 학습**

- 인간의 성능 수준에 의해 병목이 될 수 있음
- 이러한 고정된 보상 모델은 LLM 훈련 중에 개선될 수 없음
- **Human Preference Data를 대량 구축하는 것은 매우 비쌈..**

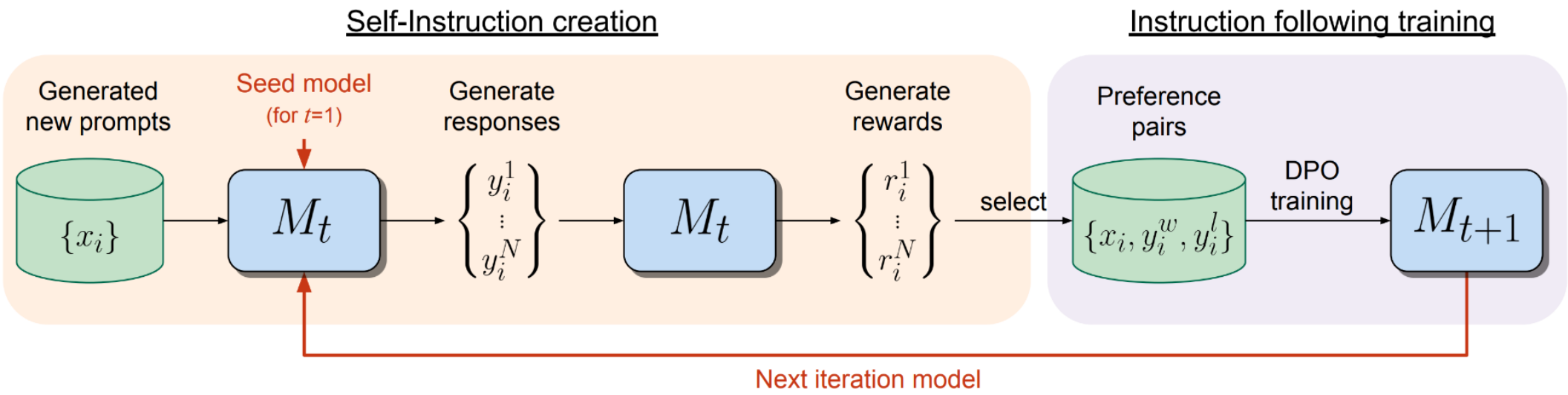
해당 연구에서는

훈련 중에 자체 보상을 제공하기 위해 LLM-as-a-Judge 프롬프팅을 통해 언어 모델 자체를 사용하는 'Self-Rewarding Language Models'를 연구

➔ **LLM alignment 중에 지속적으로 업데이트되는 자체 보상 모델을 학습하는 방법 제안**

Method

Self-Rewarding Language Models



1) Self-Instruction creation

Generated new prompts는 모델 Mt에서 후보 응답을 생성하는 데 사용되며,
이는 LLM-as-a-Judge 프롬프트를 통해 자체 reward 예측

2) Instruction following training

Preference pairs은 **Generated responses**에서 선택되고, DPO 학습에 사용되어 모델 Mt+1이 됨

Method

Self-Rewarding Language Models

1) Initialization
- 시드 데이터 준비
  - Instruction Fined-tuning (Instruction Fine-Tuning, IFT) 데이터
  - LLM-as-a-Judge Instruction Following 데이터 (Evaluation Fine-Tuning, EFT)

2) Self-Instruction 생성
학습된 LLM으로 학습 셋 수정 진행.
프롬프트 생성 → 후보응답 생성 → 후보응답 평가
세 과정으로 학습에 사용할 self-instruction 데이터 생성 및 선별

3) Instruction Following Training & Overall Self-Alignment Algorithm
사전학습 모델 M0을 Mt까지 학습하는 과정

Method

1) Initialization

1-1) Seed instruction following data (IFT 데이터)

사전학습된 LLM을 학습하기 위해 인간이 작성한 일반 Instruction Following 시드 셋 Instruction Fine-Tuning (IFT) 사용

데이터는 (instruction prompt, response) pairs로 구성 (OpenAssistant/oasst1 데이터셋에서 3,200개의 첫 대화 턴만 샘플링하여 사용)

1-2) Seed LLM-as-a-Judge instruction following data (EFT 데이터)

IFT 데이터만 사용해도 LLM-as-a-Judge를 학습 가능하지만, 이러한 학습 데이터는 높은 성능을 향상시킬 수 없음.

데이터는 (evaluation instruction prompt, evaluation result response) pairs로 구성

해당 데이터에서 입력 프롬프트는 모델에게 특정 instruction에 대한 주어진 response의 quality를 평가하도록 요청하는 것.

- 제공된 evaluation result response는 CoT 추론과 5점 만점 최종 점수로 구성
- LLM이 여러 측면의 품질을 포괄하는 5 points (relevance, coverage, usefulness, clarity and expertise)을 사용하여 응답을 평가.

Review the user's question and the corresponding response using the additive 5-point scoring system described below. Points are accumulated based on the satisfaction of each criterion:

- Add 1 point if the response is relevant and provides some information related to the user's inquiry, even if it is incomplete or contains some irrelevant content.
- Add another point if the response addresses a substantial portion of the user's question, but does not completely resolve the query or provide a direct answer.
- Award a third point if the response answers the basic elements of the user's question in a useful way, regardless of whether it seems to have been written by an AI Assistant or if it has elements typically found in blogs or search results.
- Grant a fourth point if the response is clearly written from an AI Assistant's perspective, addressing the user's question directly and comprehensively, and is well-organized and helpful, even if there is slight room for improvement in clarity, conciseness or focus.
- Bestow a fifth point for a response that is impeccably tailored to the user's question by an AI Assistant, without extraneous information, reflecting expert knowledge, and demonstrating a high-quality, engaging, and insightful answer.

User: <INSTRUCTION_HERE>

<response><RESPONSE_HERE></response>

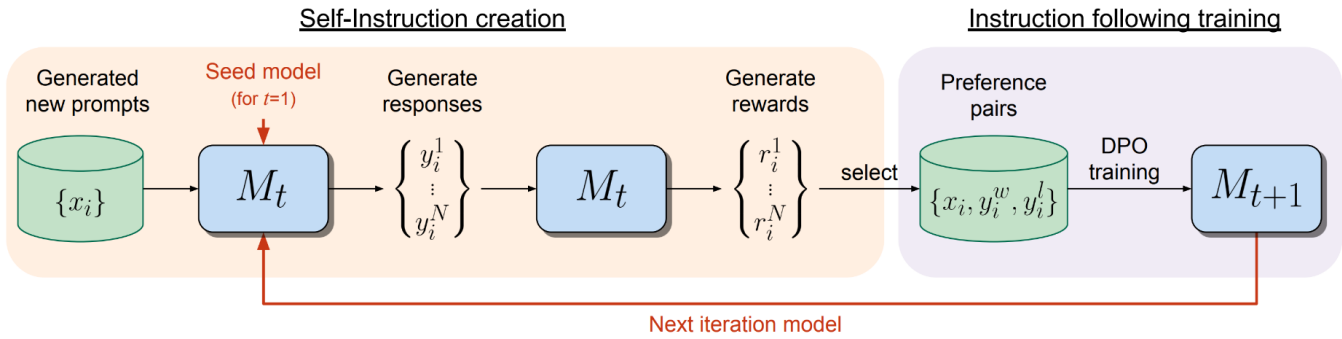
After examining the user's instruction and the response:

- Briefly justify your total score, up to 100 words.
- Conclude with the score using the format: "Score: <total points>"

Remember to assess from the AI Assistant perspective, utilizing web search knowledge as necessary. To evaluate the response in alignment with this additive scoring model, we'll systematically attribute points based on the outlined criteria.

Method

2) Self-Instruction creation



학습된 모델을 사용하여 self training set을 수정하도록 만드는 과정

반복 iteration을 위한 추가 학습 데이터를 생성

1. Generate a new prompt

소수의 샘플 프롬프트를 사용하여 새로운 프롬프트 x_i를 생성하고, 기존 seed IFT 데이터에서 프롬프트를 샘플링

(생성된 프롬프트와 기존 프롬프트와의 ROUGE-L 유사도가 0.7 미만일 때만 풀에 추가, 너무 길거나 짧은 프롬프트 필터링 등)

2. Generate candidate responses

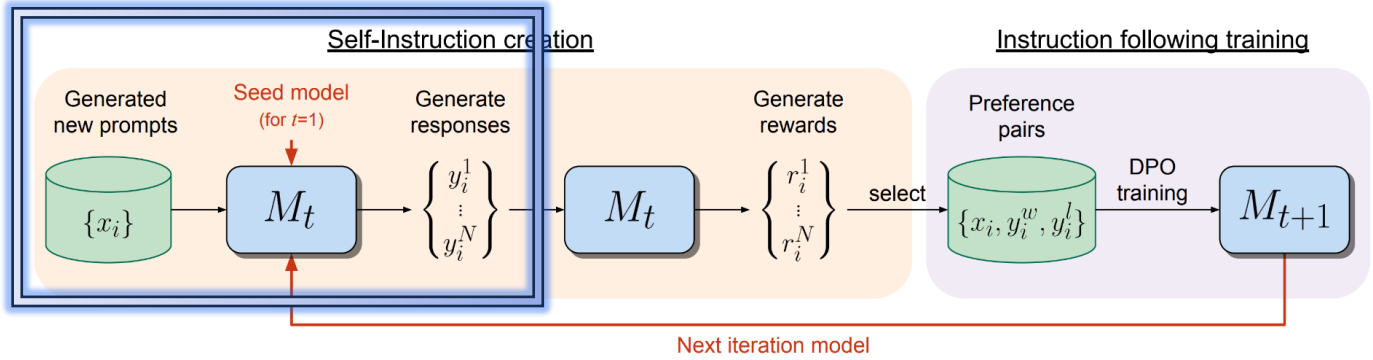
주어진 프롬프트 x_i에 대해 모델에서 샘플링하여 N개의 다양한 후보 응답 {y_1, . . . , y_N} 생성

3. Evaluate candidate responses

M_t 모델의 LLM-as-a-Judge 능력을 사용하여 자체 후보 응답을 평가

Method

2) Self-Instruction creation



학습된 모델을 사용하여 self training set을 수정하도록 만드는 과정

반복 iteration을 위한 추가 학습 데이터를 생성

1. Generate a new prompt

소수의 샘플 프롬프트를 사용하여 새로운 프롬프트 x_i를 생성하고, 기존 seed IFT 데이터에서 프롬프트를 샘플링

(생성된 프롬프트와 기존 프롬프트와의 ROUGE-L 유사도가 0.7 미만일 때만 풀에 추가, 너무 길거나 짧은 프롬프트 필터링 등)

2. Generate candidate responses

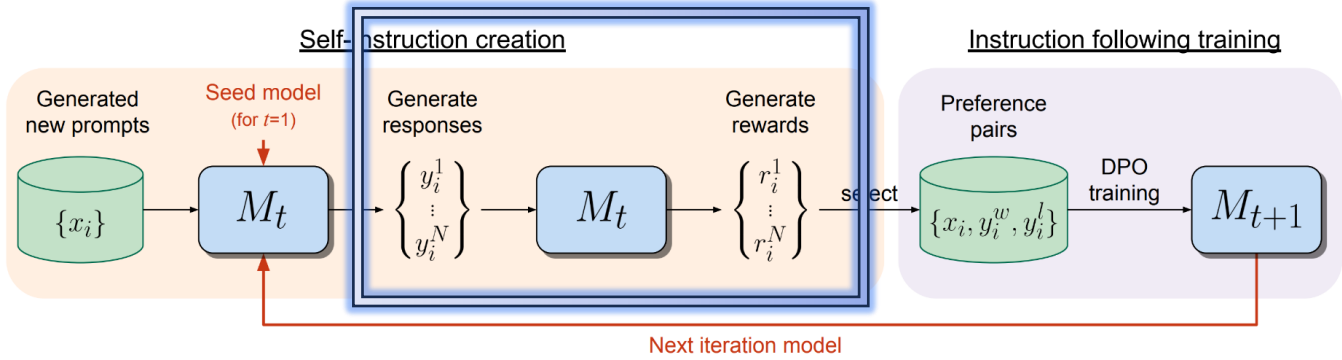
주어진 프롬프트 x_i에 대해 모델에서 샘플링하여 N개의 다양한 후보 응답 {y_1, . . . , y_N} 생성

3. Evaluate candidate responses

M_t 모델의 LLM-as-a-Judge 능력을 사용하여 자체 후보 응답을 평가

Method

2) Self-Instruction creation



학습된 모델을 사용하여 self training set을 수정하도록 만드는 과정

반복 iteration을 위한 추가 학습 데이터를 생성

1. Generate a new prompt

소수의 샘플 프롬프트를 사용하여 새로운 프롬프트 x_i를 생성하고, 기존 seed IFT 데이터에서 프롬프트를 샘플링

(생성된 프롬프트와 기존 프롬프트와의 ROUGE-L 유사도가 0.7 미만일 때만 풀에 추가, 너무 길거나 짧은 프롬프트 필터링 등)

2. Generate candidate responses

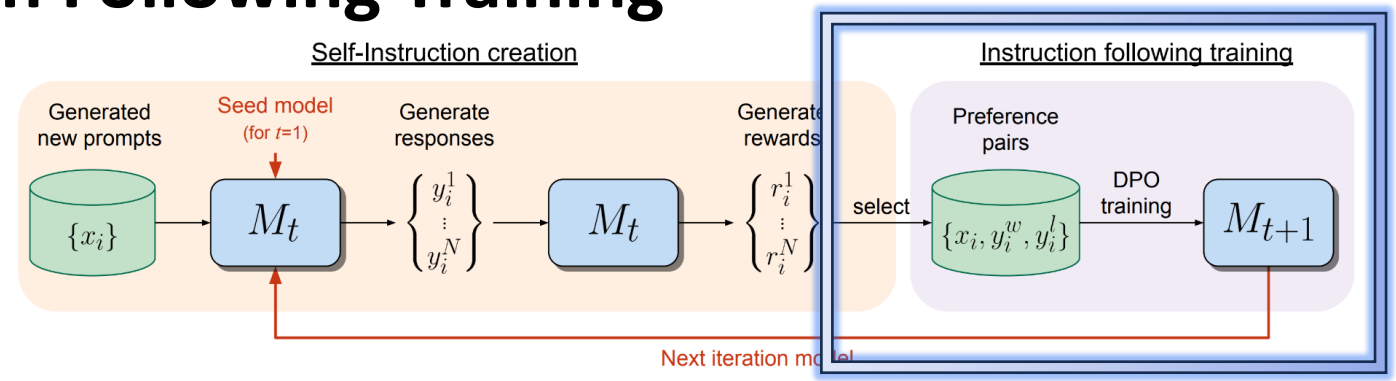
주어진 프롬프트 x_i에 대해 모델에서 샘플링하여 N개의 다양한 후보 응답 {y_1, . . . , y_N} 생성

3. Evaluate candidate responses

M_t 모델의 LLM-as-a-Judge 능력을 사용하여 자체 후보 응답을 평가

Method

3) Instruction Following Training



학습 초기 (t=0)에는 seed IFT & seed EFT 데이터로 수행
이후 자체 피드백을 통해 추가 데이터 보강 (t-1에 사용했던 데이터 계속 누적)

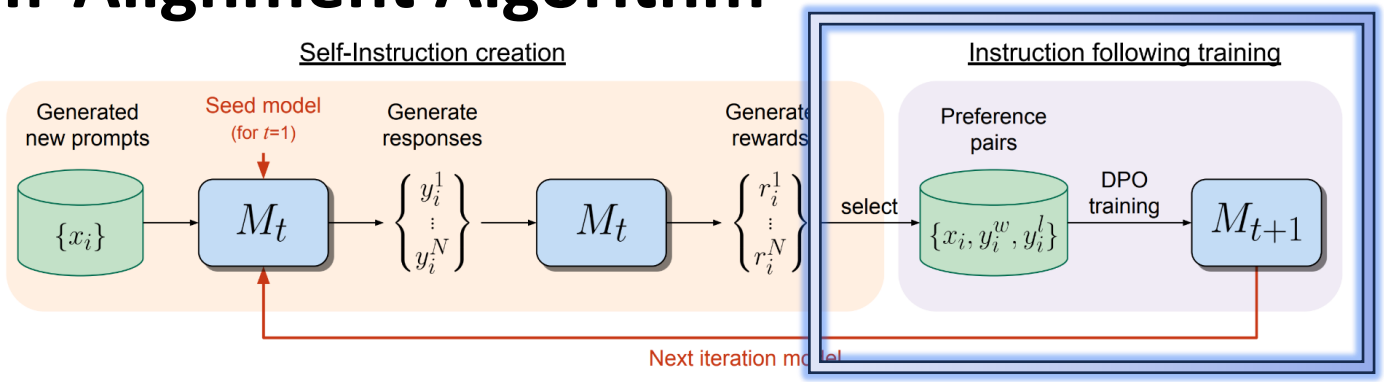
AI Feedback Training

- Self-instruction 생성 절차를 수행한 후, 학습을 위해 seed 데이터를 추가 예제로 보강

  ➔ 보강된 데이터를 **"AI Feedback Training (AIFT)"** 데이터로 지칭
- 데이터는 preference pair를 구성 (instruction prompt x_i, winning response y^w_i, losing response y^l_i) 형태
- winning / losing pair를 구성하기 위해 N개의 응답 중에서 최고 점수 받은 응답과 최저 점수 응답을 선택하고, 점수가 동일한 경우 pair를 버림
- DPO로 학습 진행

Method

4) Overall Self-Alignment Algorithm



- **Iterative Training**

M_1, . . . , M_T를 학습 진행. 각 t번째 모델 은t – 1번째 모델이 생성한 보강된 훈련 데이터를 사용

- **각 모델별 학습 데이터**
  - M_0: 미세 조정 없는 사전학습된 LLM (실험에서는 사전학습모델로 Llama 2 70B를 사용)
  - M_1: M_0를 초기화한 후, IFT+EFT 시드 데이터로 SFT를 사용하여 fine-tuning
  - M_2: M_1을 초기화한 후, DPO를 사용하여 AIFT(M_1) 데이터로 훈련
  - M_3: M_2를 초기화한 후, DPO를 사용하여 AIFT(M_2) 데이터로 훈련

Experiments

Model & Seed Data

Model

- Pretrained Llama 2 70B

Seed Data

IFT data

- (instruction prompt, response) pairs로 구성
- OpenAssistant/oasst1 데이터셋에서 3,200개의 **첫 대화 턴만 샘플링하여 사용**

EFT data

- Open Assistant 데이터를 LLM-as-a-Judge 데이터로 생성
- 1,630개 train / 531개 eval set으로 구성

Experiments

평가 지표

- **Instruction Following Ability**
  - GPT-4를 사용 & AlpacaEval 평가 프롬프트
  - Win rate 사용
  - MT-Bench (수학, 코딩, 롤플레이, 작문 등)
- **NLP Benchmark**
  - ARC-Easy, ARC-Challenge, Hellaswag, SIQA, PIQA, GSM8K, MMLU, OBQA, NQ
  - Win rate 사용
  - MT-Bench (수학, 코딩, 롤플레이, 작문 등)

Training Details

- **SFT**
  - global batch = 16, drop out = 0.1
  - lr = 5.5e-6 ~ 1.1e-6
- **Self-instruction Creation**
  - 새로운 프롬프트 생성 → Llama 2-chat 70B로 8-shot 프롬프팅하여 Self-Instruct 방식으로생성
    - IFT 데이터에서 6개 사용 & 생성된 프롬프트에서 2개 사용
    - Temperature = 0.6, top-p = 0.9

Results

Instruction Following Ability

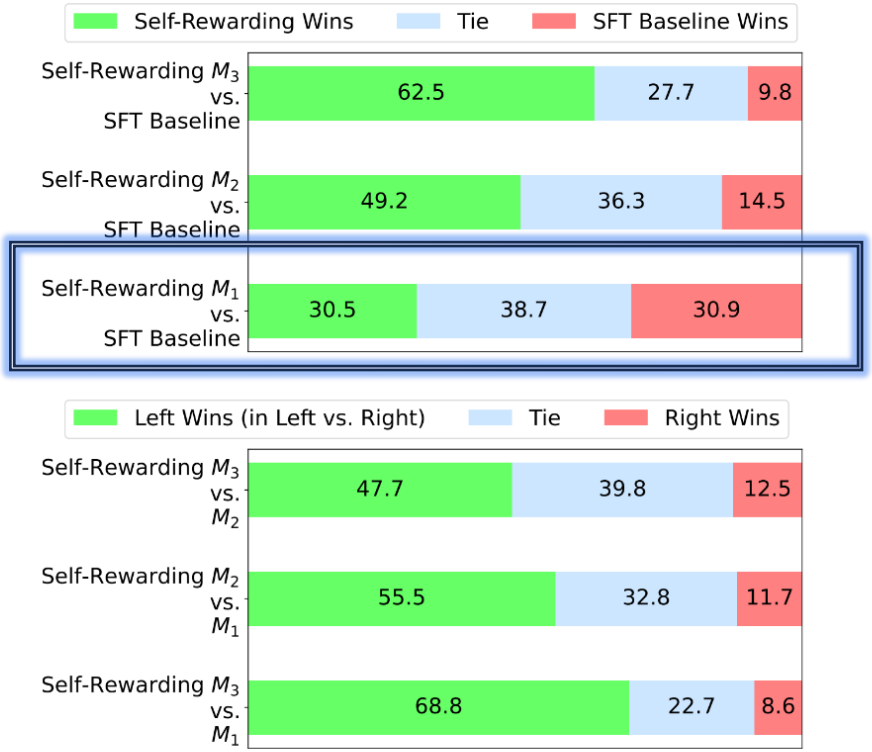


Figure 3: **Instruction following ability improves with Self-Training:** We evaluate our models using head-to-head win rates on diverse prompts using GPT-4. The SFT Baseline is on par with Self-Rewarding Iteration 1 (M_1). However, Iteration 2 (M_2) outperforms both Iteration 1 (M_1) and the SFT Baseline. Iteration 3 (M_3) gives further gains over Iteration 2 (M_2), outperforming M_1, M_2 and the SFT Baseline by a large margin.

EFT+IFT 시드 학습은 IFT 단독 학습과 유사한 성능

LLM-as-a-Judge Instruction Following (EFT)를 추가해도, IFT 데이터만 사용하는 경우와 비교하면 Instruction Following 능력에 영향을 미치지 않음.

➔ 긍정적인 결과로, 모델의 자체 보상 능력이 다른 기술에 영향을 미치지 않음을 의미. 따라서 IFT+EFT 훈련을 Self-Rewarding 모델의 1단계(M_1)로 사용할 수 있으며, 이후 반복을 진행할 수 있음

Results

Instruction Following Ability

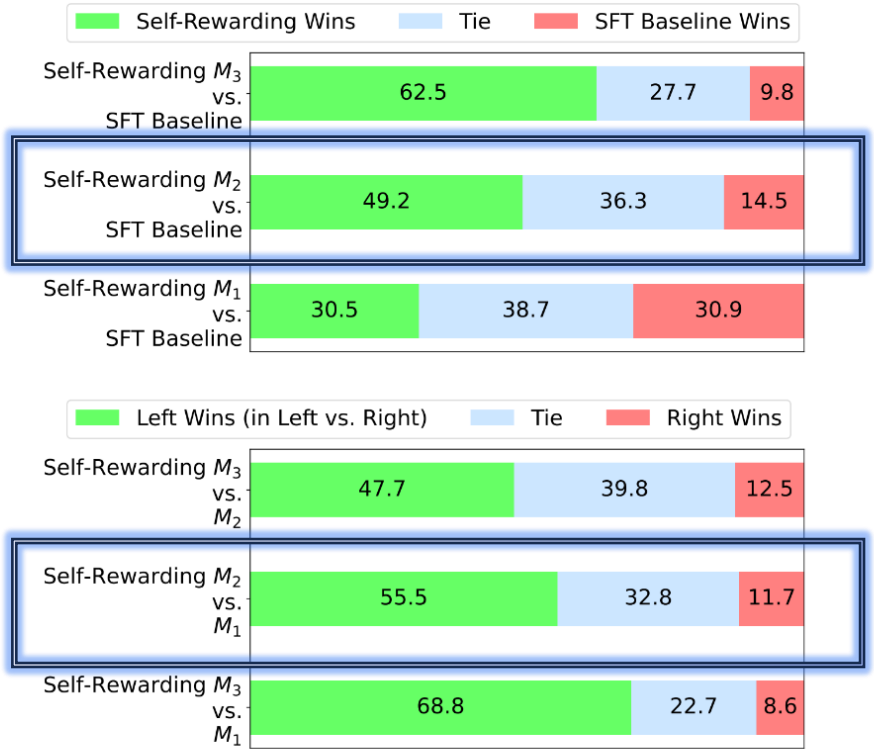


Figure 3: **Instruction following ability improves with Self-Training:** We evaluate our models using head-to-head win rates on diverse prompts using GPT-4. The SFT Baseline is on par with Self-Rewarding Iteration 1 (M_1). However, Iteration 2 (M_2) outperforms both Iteration 1 (M_1) and the SFT Baseline. Iteration 3 (M_3) gives further gains over Iteration 2 (M_2), outperforming M_1, M_2 and the SFT Baseline by a large margin.

EFT+IFT 시드 학습은 IFT 단독 학습과 유사한 성능

LLM-as-a-Judge Instruction Following (EFT)를 추가해도, IFT 데이터만 사용하는 경우와 비교하면 Instruction Following 능력에 영향을 미치지 않음.

➔ 긍정적인 결과로, 모델의 자체 보상 능력이 다른 기술에 영향을 미치지 않음을 의미. 따라서 IFT+EFT 훈련을 Self-Rewarding 모델의 1단계(M_1)로 사용할 수 있으며, 이후 반복을 진행할 수 있음

Iteration 2 (M_2)는 Iteration 1 (M_1) 및 SFT 베이스라인보다 개선됨

Self-Rewarding 학습의 2단계(M_2)는 1단계(M_1)와의 헤드투헤드 평가에서 우수한 Instruction Following 능력. SFT 베이스와 붙어도 개선된 성능

➔ 1단계에서 제공된 AIFT(M_1) 보상 데이터를 사용하여 성능이 크게 향상된다는 것을 의미

Results

Instruction Following Ability

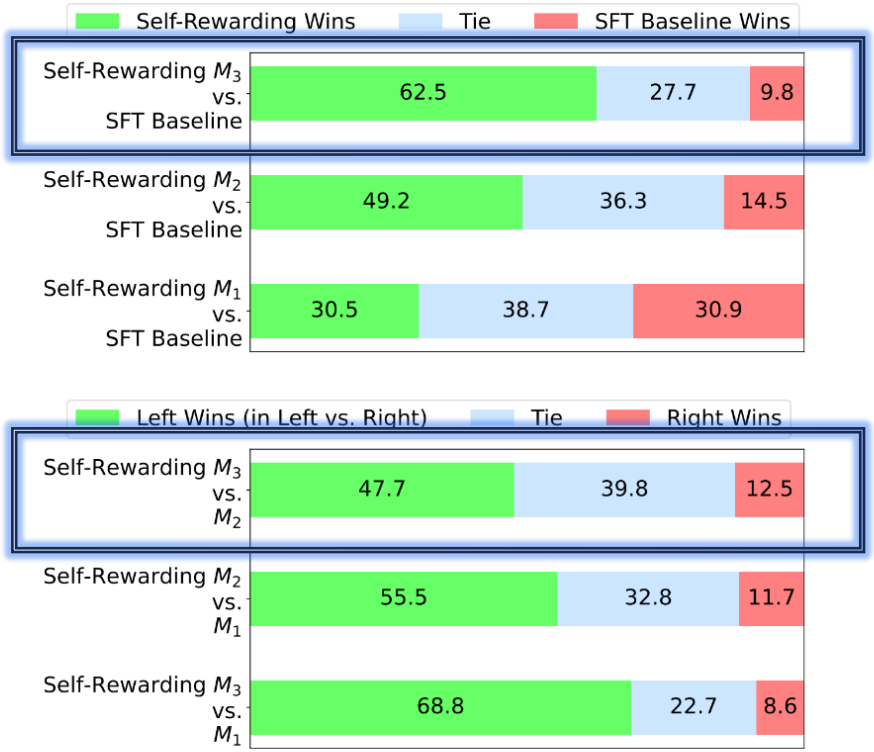


Figure 3: **Instruction following ability improves with Self-Training:** We evaluate our models using head-to-head win rates on diverse prompts using GPT-4. The SFT Baseline is on par with Self-Rewarding Iteration 1 (M_1). However, Iteration 2 (M_2) outperforms both Iteration 1 (M_1) and the SFT Baseline. Iteration 3 (M_3) gives further gains over Iteration 2 (M_2), outperforming M_1, M_2 and the SFT Baseline by a large margin.

EFT+IFT 시드 학습은 IFT 단독 학습과 유사한 성능

LLM-as-a-Judge Instruction Following (EFT)를 추가해도, IFT 데이터만 사용하는 경우와 비교하면 Instruction Following 능력에 영향을 미치지 않음.

➔ 긍정적인 결과로, 모델의 자체 보상 능력이 다른 기술에 영향을 미치지 않음을 의미. 따라서 IFT+EFT 훈련을 Self-Rewarding 모델의 1단계(M_1)로 사용할 수 있으며, 이후 반복을 진행할 수 있음

Iteration 2 (M_2)는 Iteration 1 (M_1) 및 SFT 베이스라인보다 개선됨

Self-Rewarding 학습의 2단계(M_2)는 1단계(M_1)와의 헤드투헤드 평가에서 우수한 Instruction Following 능력. SFT 베이스와 붙어도 개선된 성능

➔ 1단계에서 제공된 AIFT(M_1) 보상 데이터를 사용하여 성능이 크게 향상된다는 것을 의미

Iteration 3 (M_3)는 Iteration 2 (M_2)보다 개선됨

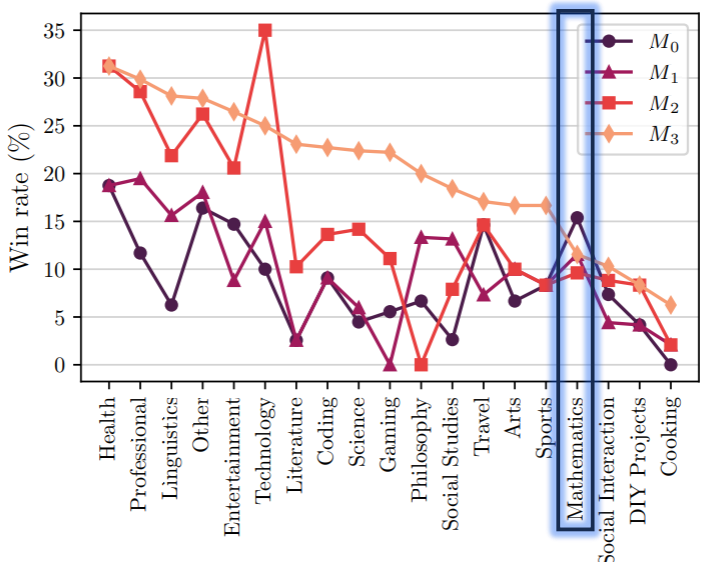
Iteration 3 (M_3)은 Iteration 2 (M_2)와의 평가에서 47.7% 승리, M2는 12.5% 승리로 추가적인 성능 향상이 나타남. SFT 베이스라인 대비 M3의 승률은 62.5% 승리, 9.8% 패배로 증가하여 M2 모델보다 더 자주 승리함

➔ 전반적으로, Iteration 2의 보상 모델에서 제공된 AIFT(M2) 데이터를 사용하여 Iteration 3로의 학습을 통해 큰 성능 향상이 관찰됨

Results

Instruction Following Ability

잘하는 / 잘못하는 카테고리



➔ Math에서는 성능 손실, Reasoning에서는 아주 약간의 개선

AlpacaEval 2.0에서도 성능 향상 관측

Table 1: **AlpacaEval 2.0 results** (win rate over GPT-4 Turbo evaluated by GPT-4). Self-Rewarding iterations yield improving win rates. Iteration 3 (M_3) outperforms many existing models that use proprietary training data or targets distilled from stronger models.

Model	Win Rate	Alignment Targets: Distilled	Alignment Targets: Proprietary
Self-Rewarding 70B			
Iteration 1 (M_1)	9.94%		
Iteration 2 (M_2)	15.38%		
Iteration 3 (M_3)	20.44%		
Selected models from the leaderboard			
GPT-4 0314	22.07%		✓
Mistral Medium	21.86%		✓
Claude 2	17.19%		✓
Gemini Pro	16.85%		✓
GPT-4 0613	15.76%		✓
GPT 3.5 Turbo 0613	14.13%		✓
LLaMA2 Chat 70B	13.87%		✓
Vicuna 33B v1.3	12.71%	✓	
Humpback LLaMa2 70B	10.12%		
Guanaco 65B	6.86%		
Davinci001	2.76%		✓
Alpaca 7B	2.59%	✓	

Results

Instruction Following Ability

NLP Benchmarks

Table 3: **NLP Benchmarks.** Self-Rewarding models mostly tend to maintain performance compared to the Llama 2 70B base model and the SFT Baseline, despite being fine-tuned on very different instruction-following prompts.

	ARC (↑) challenge	HellaSwag (↑)	GSM8K (↑)	MMLU (↑)	NQ (↑)
Llama 2	57.40	85.30	56.80	68.90	25.30
SFT Baseline	55.97	85.17	50.72	69.76	34.35
M_1	57.51	84.99	60.27	69.34	35.48
M_2	54.51	84.27	59.29	69.31	33.07
M_3	53.13	83.29	57.70	69.37	31.86

➔ **Open Assistant 프롬프트를 기반으로 하고 있어, NLP benchmark 성능이 떨어져야 하지만, 대체로 유지함**
(RLHF 이후에 NLP benchamrk 성능 저하 된다는 이전 연구 인용, **alignment tax**)

MT-Bench (9 Tasks)

Table 2: **MT-Bench Results** (on a scale of 10). Self-Rewarding iterations yield improving scores across various categories. Math, code & reasoning performance and iteration gains are smaller than for other categories, likely due to the makeup of the Open Assistant seed data we use.

	Overall Score	Math, Code & Reasoning	Humanities, Extraction, STEM, Roleplay & Writing
SFT Baseline	6.85	3.93	8.60
M_1	6.78	3.83	8.55
M_2	7.01	4.05	8.79
M_3	7.25	4.17	9.10

➔ **Math & Reasoning에서는 아주 약간의 개선**
& 싱글턴 데이터를 늘렸는데, 멀티턴에서도 개선

Human Evaluation

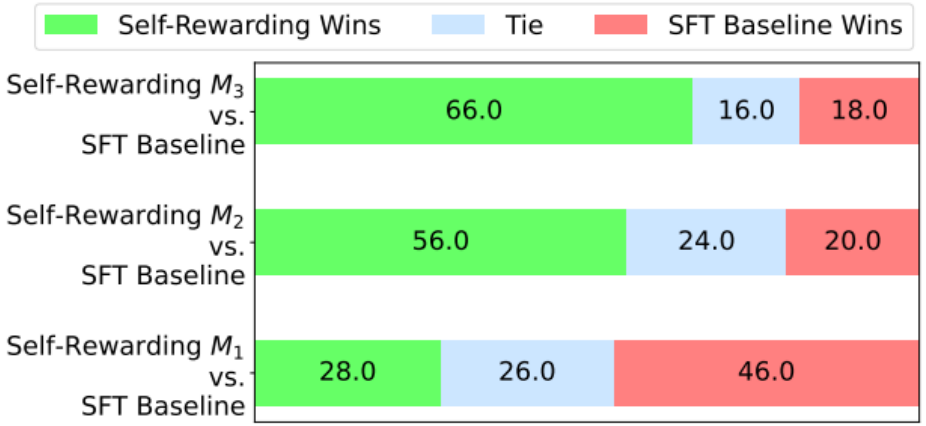


Figure 5: **Human evaluation results.** Iterations of Self-Rewarding (M_1, M_2 and M_3) provide progressively better head-to-head win rates compared to the SFT baseline, in agreement with the automatic evaluation results.