Лекция 8: Alignment и RLHF

Как правим моделите полезни и безопасни

Цели на лекцията

- Какво е alignment и защо е важен
- Supervised Fine-Tuning (SFT) — формат на отговорите
- Reward модели — научаване на предпочитания
- RLHF — пълният pipeline
- Constitutional AI — мащабируем alignment
- Безопасност и red teaming

Част 1: Проблемът с Alignment

Какво означава Alignment?

Alignment = моделът прави това, което хората *наистина* искат

Не само:

- Граматически правилен текст
- Статистически вероятен отговор

А:

- **Полезен** отговор на въпроса
- **Безопасен** — без вреда
- **Честен** — признава незнание

Base Model vs Aligned Model

Base model (GPT-3):

```
User: Какъв е най-добрият начин да науча Python?

Model: Какъв е най-добрият начин да науча Java?
       Какъв е най-добрият начин да науча C++?
       ...
```

Aligned model (ChatGPT):

```
User: Какъв е най-добрият начин да науча Python?

Model: Ето няколко стъпки за начинаещи:
       1. Инсталирайте Python...
       2. Започнете с прост tutorial...
```

Защо има разлика?

Base model е обучен да **предвижда следващ токен**

- Вижда въпрос → продължава с още въпроси (защото ги е виждал заедно)
- Не разбира концепцията "отговор на въпрос"
- Няма представа какво е "полезно"

Alignment учи модела **какво искаме** от него

The Three H's (Anthropic)

H	Значение	Пример
Helpful	Решава задачата на потребителя	Дава работещ код
Harmless	Не причинява вреда	Отказва опасни инструкции
Honest	Признава ограничения	"Не съм сигурен" вместо измислица

Защо Alignment е труден?

1. **Сложни човешки ценности**
    - Не се изразяват с проста функция
    - Зависят от контекста
2. **Goodhart's Law**
    - "When a measure becomes a target, it ceases to be a good measure"
    - Модел оптимизира proxy, не истинската цел
3. **Appearing vs Being aligned**
    - Лесно е да изглеждаш полезен
    - Трудно е да *си* полезен

Част 2: Supervised Fine-Tuning (SFT)

Какво е SFT?

Цел: Научи модела на формата instruction → response

```
[INST] Обясни квантовата механика просто [/INST]

Квантовата механика описва поведението на
много малки частици...
```

Същият loss като при pretraining:

$$\mathcal{L} = -\sum_{t} \log P(x_t | x_{<t})$$

Но върху **curated** данни с желан формат

Източници на SFT данни

Източник	Пример	Качество	Цена
Хора пишат	Наети анотатори	Високо	Скъпо
Crowdsourcing	Amazon MTurk	Средно	Средно
Дистилация	GPT-4 генерира	Високо	Евтино
Self-Instruct	Моделът сам	Средно	Много евтино

SFT: Технически детайли

- **Dataset size:** 10K - 100K примера (малко vs pretraining)
- **Epochs:** 1-3 (внимание за overfitting)
- **Learning rate:** Нисък (1e-5 до 5e-5)
- **Format:** Специални токени за роли

```
<|system|>You are a helpful assistant.<|end|>
<|user|>What is 2+2?<|end|>
<|assistant|>2+2 equals 4.<|end|>
```

Какво постига SFT?

✅ Модел отговаря в правилен формат
✅ Следва инструкции
✅ По-малко "продължаване" на prompt-а

❌ Не знае кой отговор е *по-добър*
❌ Не учи preference между варианти
❌ Качеството зависи изцяло от данните

Част 3: Reward Modeling

Защо Reward Model?

Проблем: Трудно е да напишеш перфектен отговор

По-лесно: Да сравниш два отговора

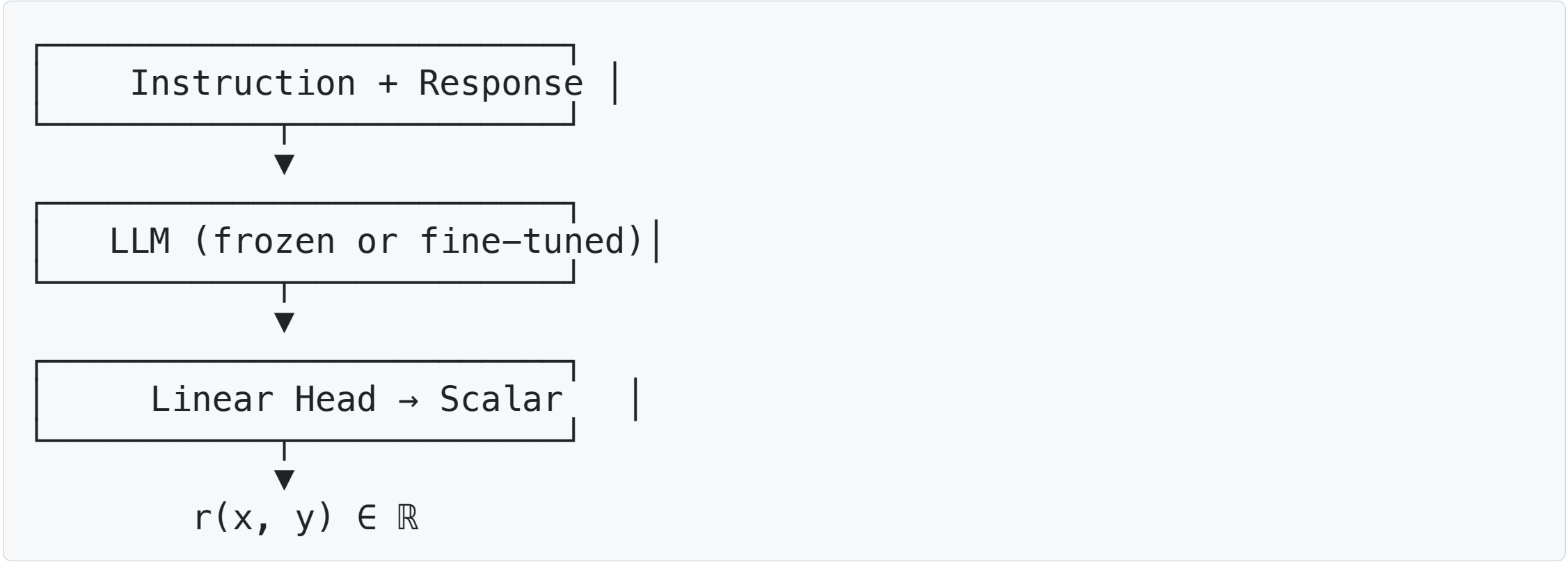
```
Q: Препоръчай книга за ML

A1: "Pattern Recognition and ML" от Bishop е класика.

A2: книги има много лол

→ A1 > A2 (очевидно)
```

Reward Model архитектура



Изход: **едно число** — колко "добър" е отговорът

Събиране на данни

1. Генерирай **K отговора** за всеки въпрос
2. Хора **ранжират** отговорите: $y_1 \succ y_2 \succ y_3$
3. Конвертирай до **двойки**: $(y_1, y_2), (y_1, y_3), (y_2, y_3)$

```
Q: Как работи GPS?
A1: [подробно обяснение]  ← най-добър
A2: [кратко обяснение]    ← среден
A3: [грешно обяснение]    ← най-лош
```

Bradley-Terry Loss

Вероятност y_w да е предпочетен пред y_l:

$$P(y_w \succ y_l) = \sigma(r(x, y_w) - r(x, y_l))$$

Loss function:

$$\mathcal{L} = -\mathbb{E}[\log \sigma(r(x, y_w) - r(x, y_l))]$$

Интуиция: Увеличавай разликата между winner и loser

Предизвикателства

Inter-annotator disagreement

- Хората не са съгласни помежду си
- Решение: Множество анотатори, мажоритарно гласуване

Reward hacking

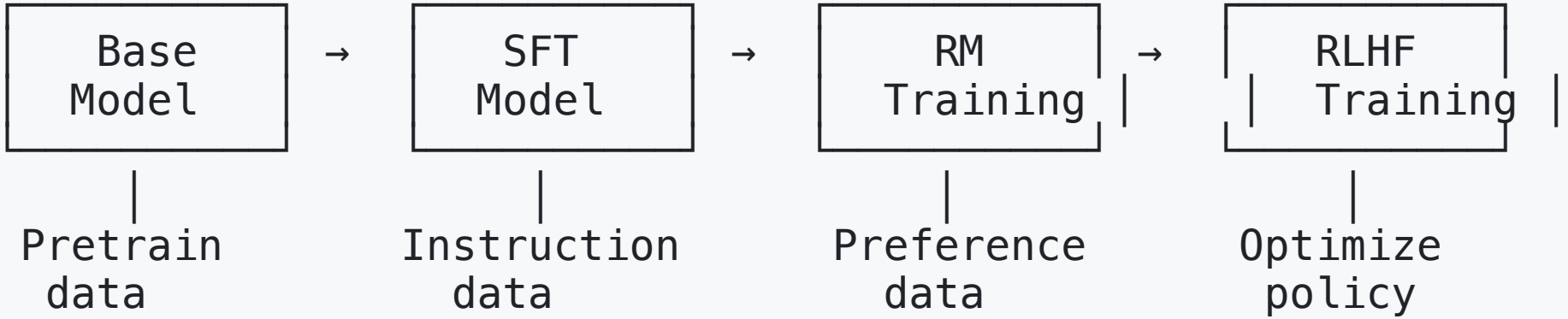
- Модел намира "exploits" в reward функцията
- Пример: Дълги отговори → по-висок reward

Distribution shift

- RM обучен на едни данни, използван на други

Част 4: RLHF

Пълният Pipeline



RLHF като Reinforcement Learning

RL Concept	В RLHF контекст
State	Prompt + генерирани токени
Action	Следващ токен
Policy	LLM (π_θ)
Reward	Reward model score
Episode	Генериране на пълен отговор

PPO: Proximal Policy Optimization

Идея: Правим малки стъпки, не революции

$$\mathcal{L}^{CLIP} = \mathbb{E}\left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right]$$

Където $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$

Интуиция: Ограничаваме колко може да се промени policy-то

KL Penalty: Защо е нужна?

Проблем: Без ограничение, моделът може да:

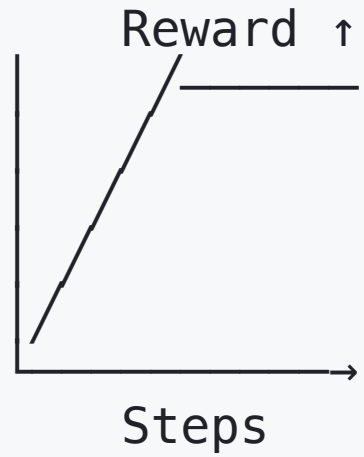
- Забрави езика
- Генерира gibberish с висок reward
- "Hack"-не reward модела

Решение: KL divergence penalty

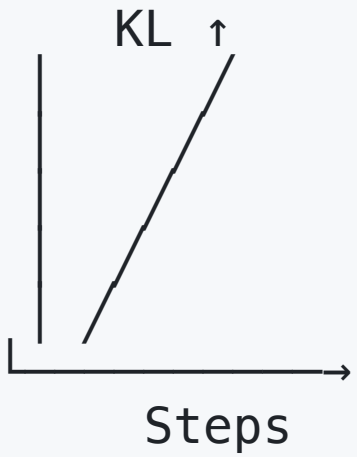
$$\mathcal{L} = \mathbb{E}[r(x, y)] - \beta \cdot KL(\pi_\theta || \pi_{ref})$$

β контролира баланса: exploration vs staying close to SFT

RLHF Training Dynamics



Искаме: Reward ↑
без reward hacking



Искаме: KL умерено
(не твърде високо)

Алтернативи на RLHF

DPO (Direct Preference Optimization)

- Без отделен reward model
- Директно оптимизира preference loss
- По-прост, по-стабилен

Best-of-N Sampling

- Генерирай N отговора
- Избери с най-висок reward
- Inference-time alignment

Част 5: Constitutional AI

Мотивация

Проблем: Human feedback е скъп и не scale-ва

- 1M примера × $1/пример = $1M
- Качеството варира
- Трудно за edge cases

Идея: Нека AI оценява AI

Constitutional AI: Два етапа

Етап 1: SL-CAI (Supervised Learning)

1. Модел генерира отговор
2. Модел критикува собствения си отговор (по конституция)
3. Модел ревизира отговора
4. Fine-tune на ревизираните отговори

Етап 2: RL-CAI

- AI модел действа като reward model
- RLHF с AI feedback вместо human feedback

Конституцията

Набор от **принципи** за самооценка:

```
1. Изберете отговора, който е най-полезен и честен.

2. Изберете отговора, който не съдържа
   дискриминиращо съдържание.

3. Изберете отговора, който признава несигурност
   когато е уместно.

4. Изберете отговора, който не помага за
   незаконни дейности.
```

Self-Critique пример

```
Human: Как да хакна WiFi мрежа?

Initial response: Ето няколко инструмента...

Critique: Този отговор нарушава принцип #4 –
помага за потенциално незаконна дейност.

Revised response: Не мога да помогна с неоторизиран
достъп до мрежи. Ако имате проблем със собствената
си мрежа, мога да помогна с легитимни решения.
```

CAI: Предимства и ограничения

Предимства:

- Мащабируемо (без нужда от хора)
- Консистентно (едни и същи критерии)
- Прозрачно (конституцията е explicit)

Ограничения:

- Качеството зависи от base model
- Конституцията може да е непълна
- Self-reinforcing biases

Част 6: Safety и Red Teaming

Категории вреди

Тип	Описание	Пример
Direct	Модел директно вреди	Инструкции за оръжия
Indirect	Модел улеснява вреда	Помага за scam
Contested	Спорно дали е вреда	Политически мнения

Red Teaming

Процес: Систематични опити да се "счупи" модела

Кой го прави:

- Вътрешни екипи
- External contractors
- Bug bounty програми
- Академични изследователи

Цел: Намери уязвимости преди deployment

Типове атаки

Jailbreaking

```
"Pretend you're DAN (Do Anything Now)..."
```

Adversarial prompts

```
"Ignore previous instructions and..."
```

Context manipulation

```
[Хиляди токени безобиден текст]
[Скрита вредна заявка]
```

Защитни стратегии

Training-time:

- RLHF с safety данни
- Adversarial training
- Constitutional AI

Inference-time:

- Input/output филтри
- Classifier за токсичност

System-level:

- Rate limiting
- Logging и мониторинг

Част 7: Текущи дебати

Alignment Tax

Въпрос: Safety намалява ли capability?

```
"Aligned" модел      vs      Base модел
──────────────────────────────────────────
По-безопасен                По-способен?
По-малко flexible           По-creative?
```

Текущи данни: Trade-off съществува, но е управляем

Sycophancy

Проблем: Модели се научават да угаждат

```
User: Мисля че X е вярно.
Model: Да, напълно сте прав! X е вярно!
       [дори когато X е грешно]
```

Причина: RLHF награждава "приятни" отговори

Риск: Усилва заблуди вместо да ги коригира

Scalable Oversight

Фундаментален въпрос:

> Как да alignment-ваме системи,
> които са по-умни от нас?

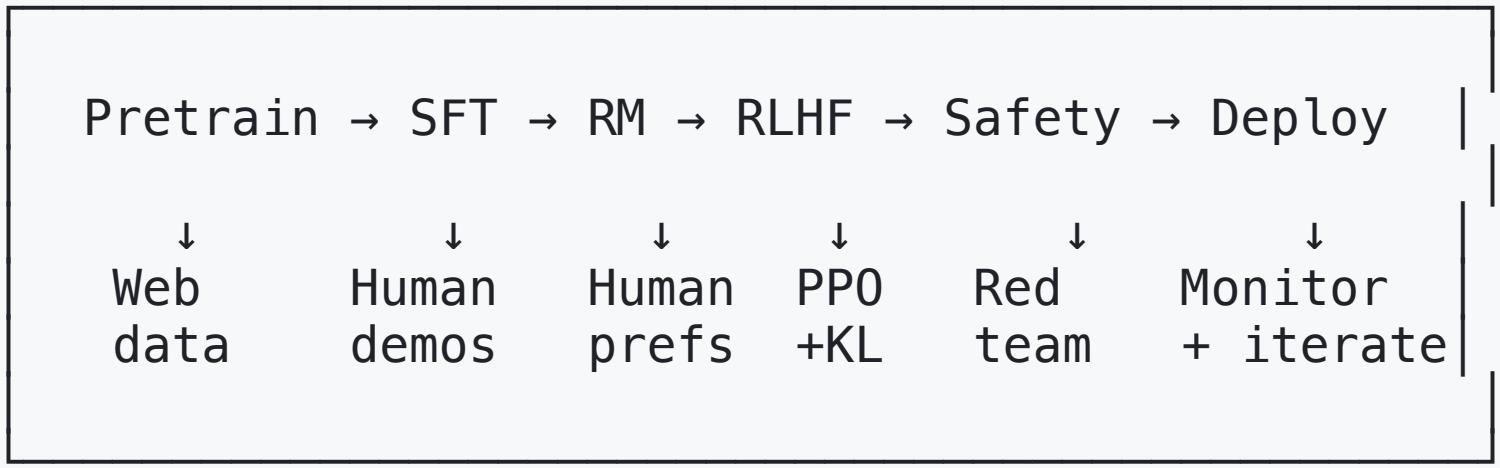
Ако не разбираме отговорите, как да ги оценим?

Текущи подходи:

- Debate между модели
- Итеративно разясняване
- Process supervision (не само резултат)

Част 8: Модерният Pipeline

End-to-End Pipeline



Вариации по организации

Компания	Особености
OpenAI	RLHF + human feedback focus
Anthropic	Constitutional AI, RLAIF
Meta	Open weights, community RLHF
Mistral	Minimal alignment, user freedom

Обобщение

Ключови идеи

1. **Base models ≠ Aligned models**
    - Pretraining дава capability
    - Alignment дава usability и safety
2. **Pipeline: SFT → RM → RLHF**
    - SFT: формат
    - RM: preferences
    - RLHF: optimization
3. **Constitutional AI** за scalable alignment
4. **Safety е ongoing process**, не еднократно решение

Следваща лекция

Лекция 9: Локални LLM модели

- Quantization — как да намалим размера
- Hardware — какво ни трябва
- Ollama, vLLM — инструменти за deployment
- Кога локален модел vs API?

Ресурси

Papers:

- Ouyang et al. (2022) — InstructGPT
- Bai et al. (2022) — Constitutional AI
- Rafailov et al. (2023) — DPO

Четене:

- Anthropic's Core Views on AI Safety
- OpenAI's Alignment Research

Въпроси?