Credit Fraud Detector

Based on RLHF

2023/06/29

Introduction

Epsilon Greedy ε would decay during learning

$$a = \begin{cases} \arg\max_a Q(s,a), & \text{with probability } 1-\varepsilon \\ random, & otherwise \end{cases}$$

上次使用 Epsilon-Greedy 時有遇到兩個問題：

1. 大部分 Explore & exploitation 方法只會取最大那群。（ex: Epsilon-Greedy 只會取 argmax）
2. 沒有讓模型參與到打分數的環節。

上禮拜的 ppt

8 - Clusters

0.05%	0.00%	86.85%	0.01%	0.00%	0.00%	71.28%	0.04%

3. 將資料轉回 PCA 前的資料（28維度），加回閾值內資料的資料集，訓練模型查看結果

4. 將Epsilon乘上decay，並更新機率表如下：

Index	0	1	2	3	4	5	6	7
P	0.0	0.0	0.8	0.0	0.0	0.0	0.4	0.0

Q: Epsilon-Greedy中，會用argmax 選擇最大的機率，代表第六群會被忽視。
第二群全部抽完的機率也有86%，所以理論上會等到完全抽完第二群才抽第六群。

Introduction

Epsilon Greedy ε would decay during learning

$$a = \begin{cases} \arg\max_a Q(s,a), & \text{with probability } 1-\varepsilon \\ random, & otherwise \end{cases}$$

上次使用 Epsilon-Greedy 時有遇到兩個問題：

1. 大部分 Explore & exploitation 方法只會取最大那群。（ex: Epsilon-Greedy 只會取 argmax）
2. 沒有讓模型參與到打分數的環節。

8 - Clusters

0.05%	0.00%	86.85%	0.01%	0.00%	0.00%	71.28%	0.04%

3. 將資料轉回 PCA 前的資料（28維度），加回閾值內資料的資料集，訓練模型查看結果

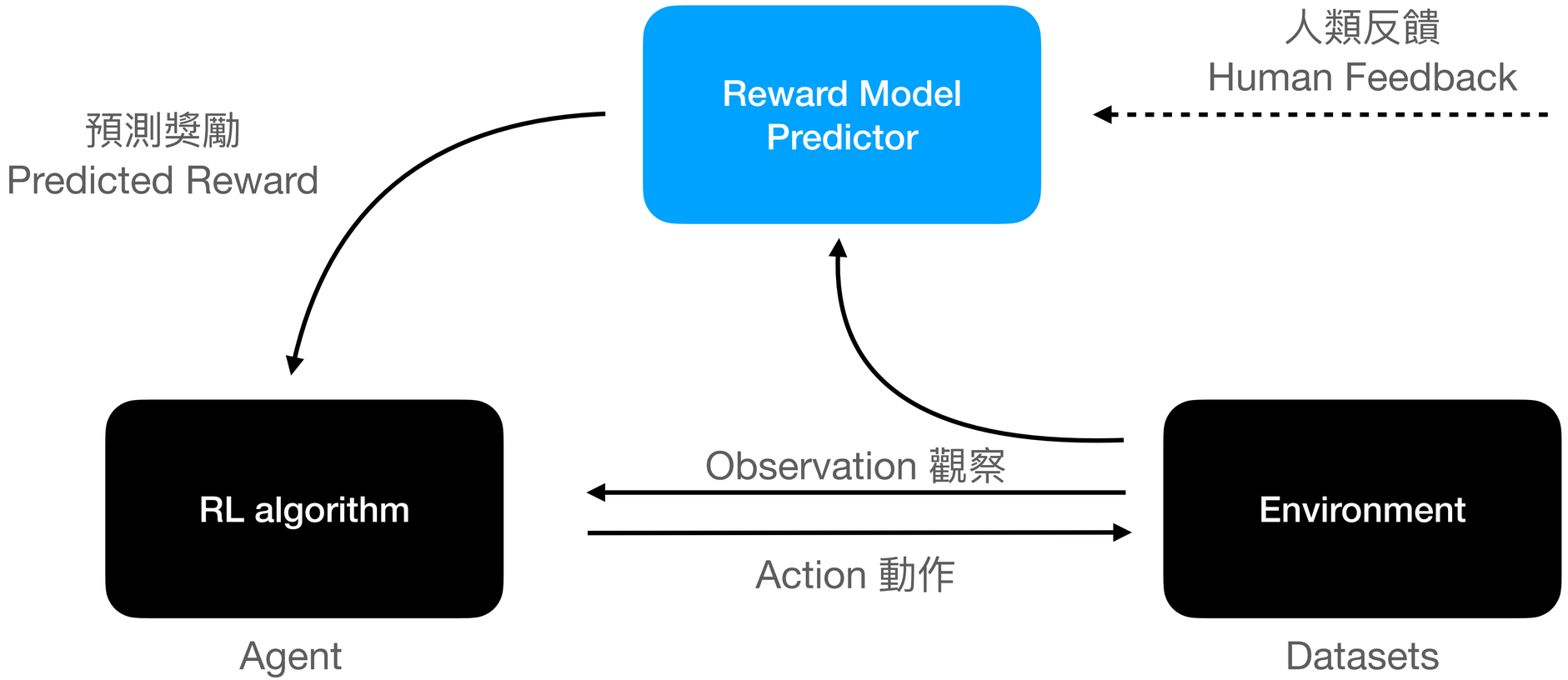
4. 將Epsilon乘上decay，並更新機率表如下：

Index	0	1	2	3	4	5	6	7
P	0.0	0.0	0.8	0.0	0.0	0.0	0.4	0.0

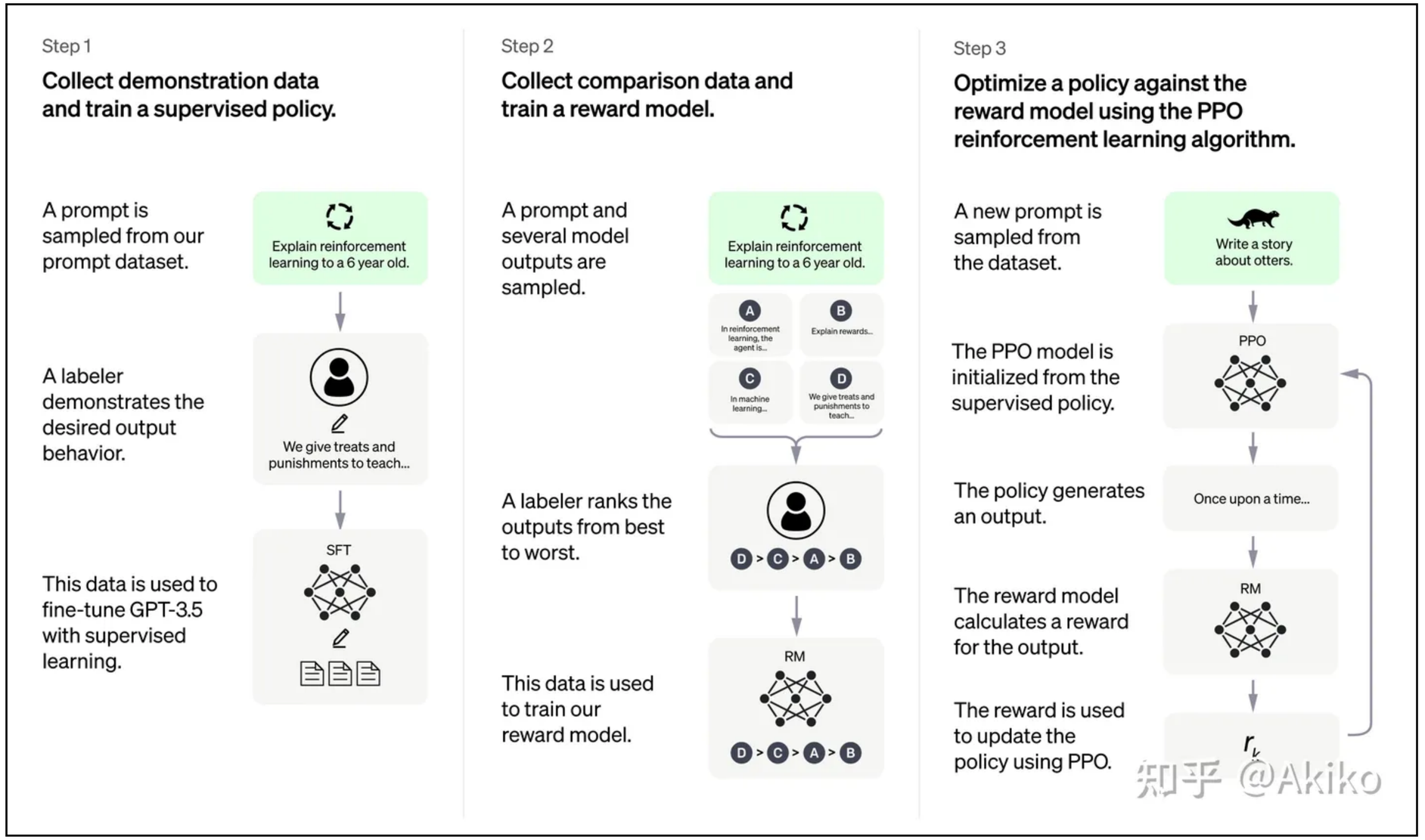
Q: Epsilon-Greedy中，會用argmax 選擇最大的機率，代表第六群會被忽視。
第二群全部抽完的機率也有86%，所以理論上會等到完全抽完第二群才抽第六群。

RLHF：從人類反饋中學習

通過人類標注數據訓練得到 Reward Model （相當於是去學人類怎麼標注資料），有了 Reward Model後，就可以使用一般的強化學習方法去找出最佳策略。

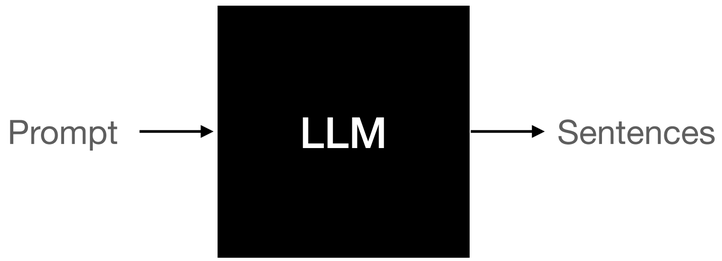


RLHF ：ChatGPT



RLHF ：ChatGPT

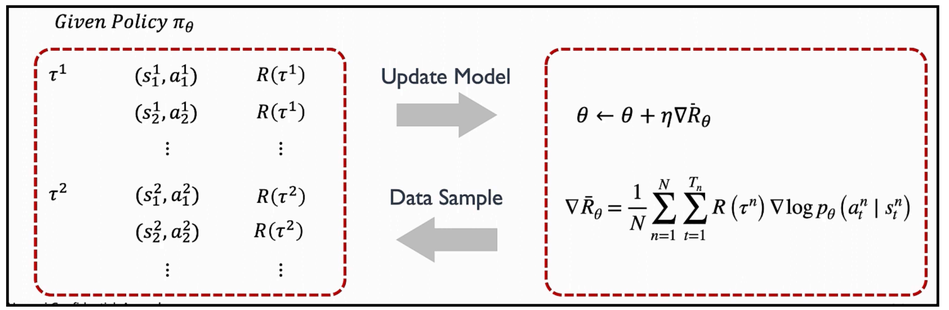
Step 1
Initial Model



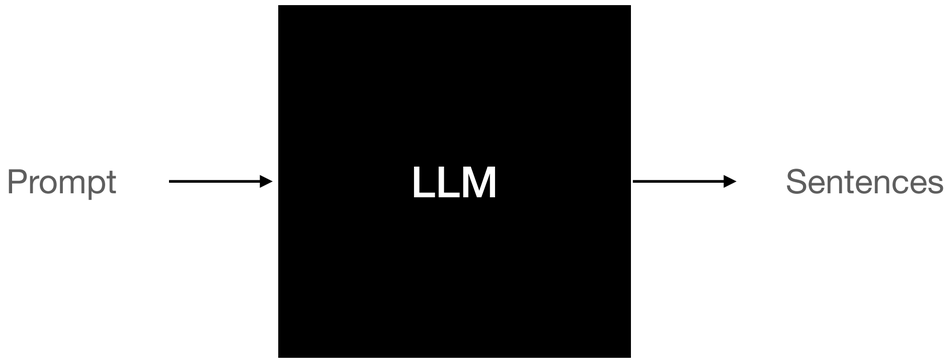
Step 2
Model Generate Sentences

Step 3
Human Feedback
學習一個 reward model

Step 4
生成句子，用Reward Model打分數，更新模型權重

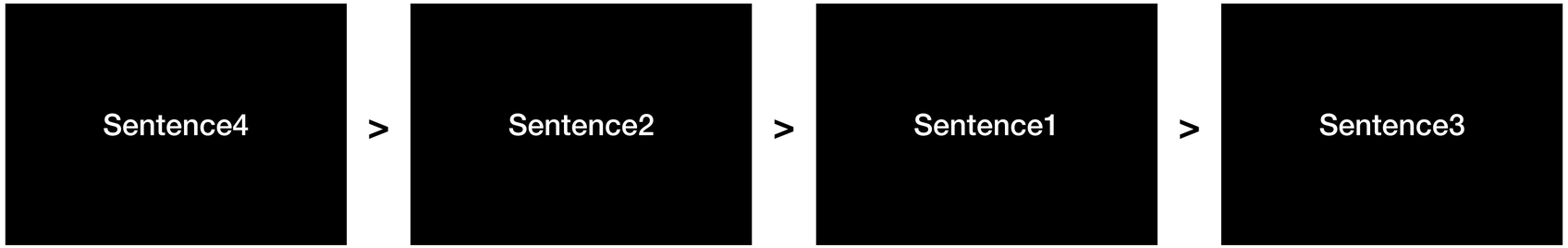


Step 1
Initial Model

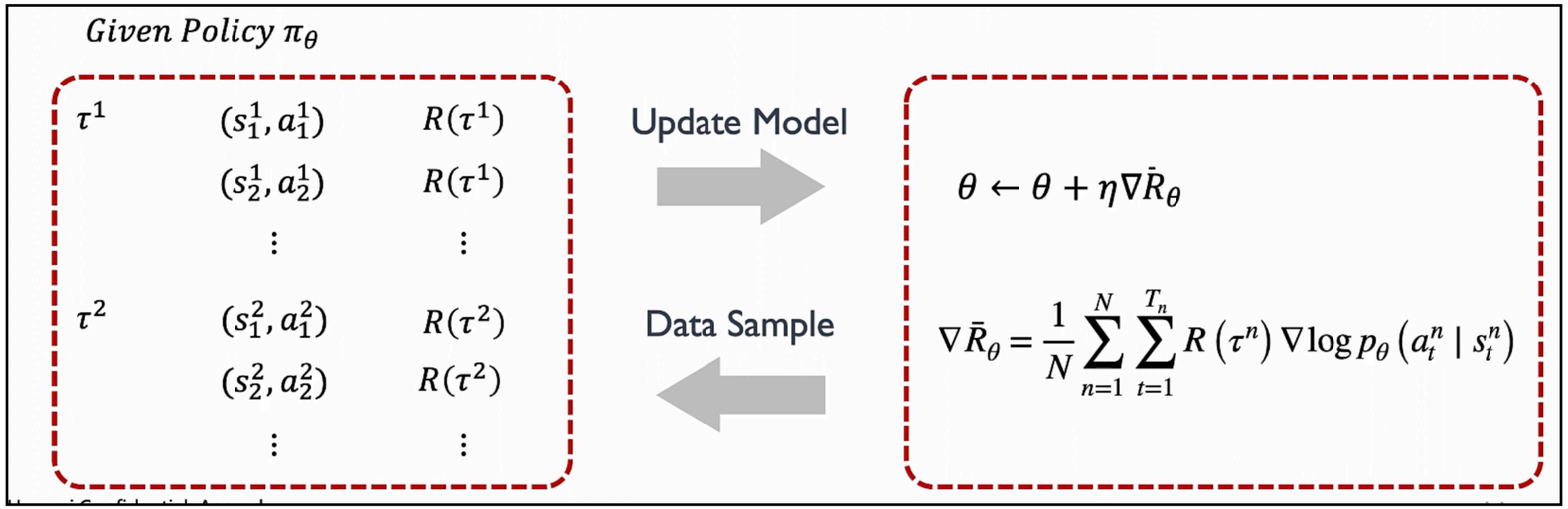


Step 2
Model Generate Sentences

Step 3
Human Feedback

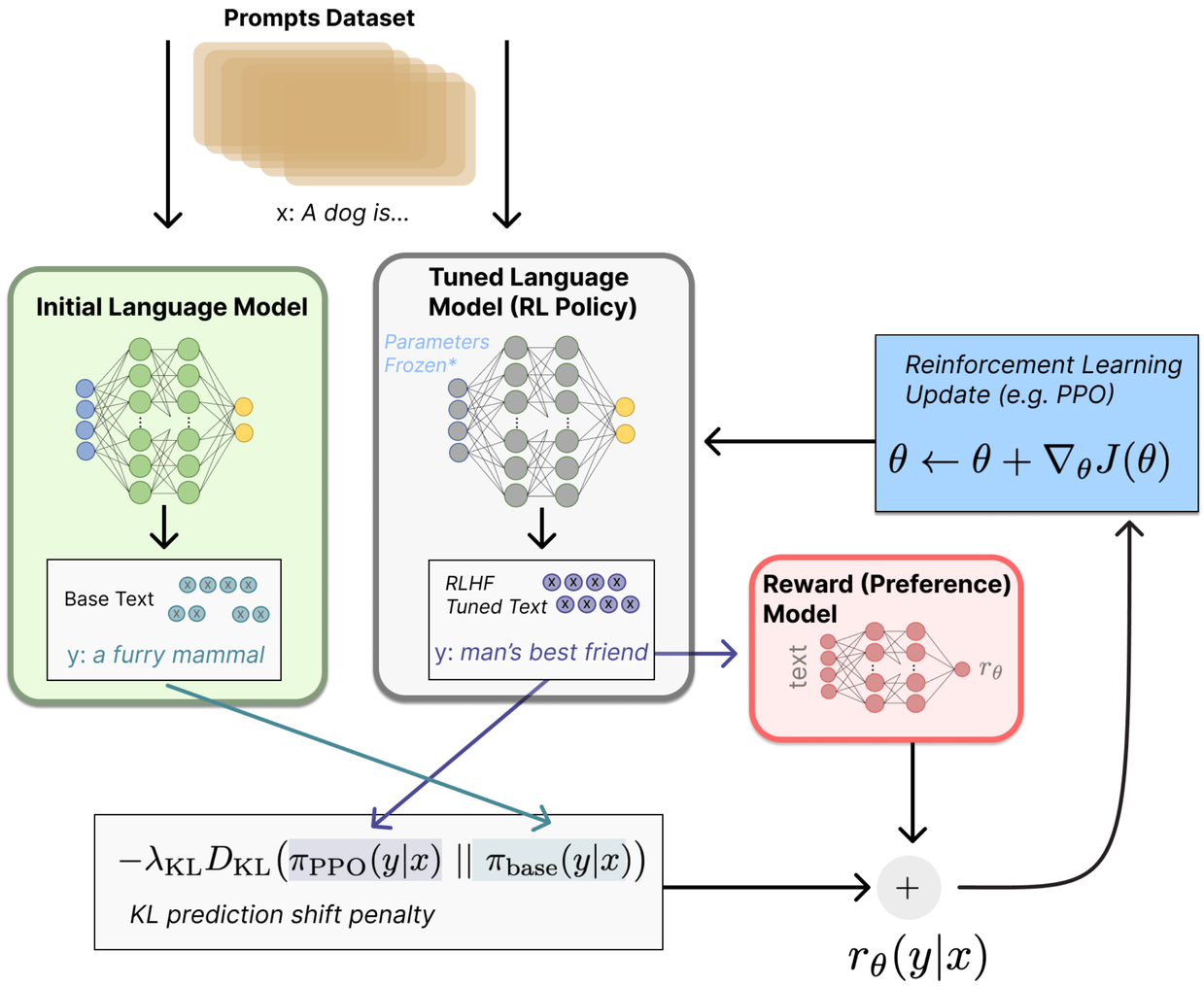


Step 4
更新模型權重



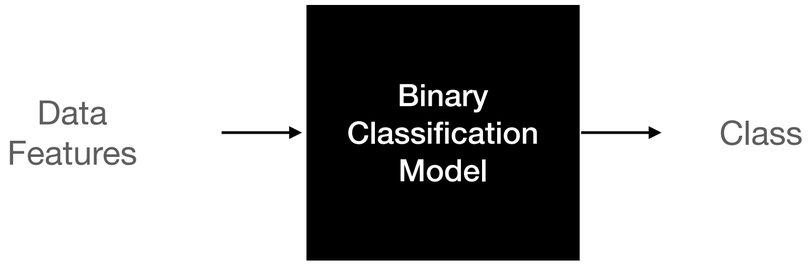
在實際環境中，讓 Agent 和 Environment互動，產生一系列採樣數據。
即獲得很多的 (s, a) 和R(t) 的 Pair 。（在狀態 s 下，採取動作 a 得到的獎勵 R(t) ）
最後將數據送到訓練過程中計算，更新模型的參數 theta。

Step 4
更新模型權重



RLHF ：OUR

Step 1
Initial Model

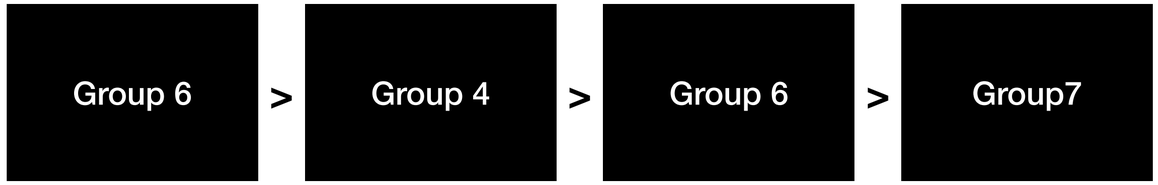


Step 2
Select Data

作法
1. PCA + Cluster 分成一些群
2. 先挑選一些群
3. 從每一群中挑選 10 筆資料

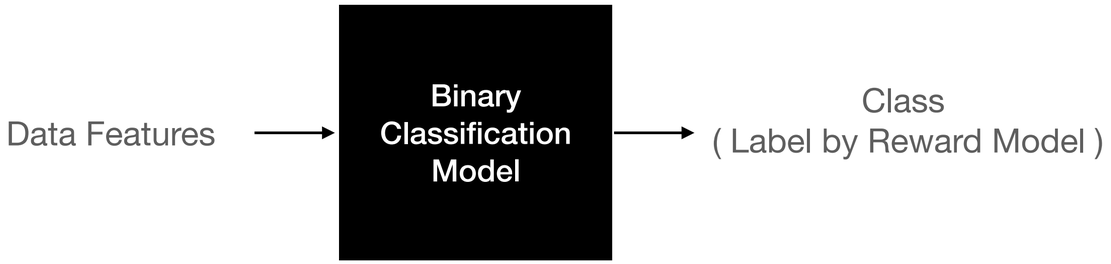
2 : 用 Epsilon-Greedy 決定要挑哪些群
3 : 用 Uncertainty Sampling 挑選資料

Step 3
Human Feedback
Training Reward Model

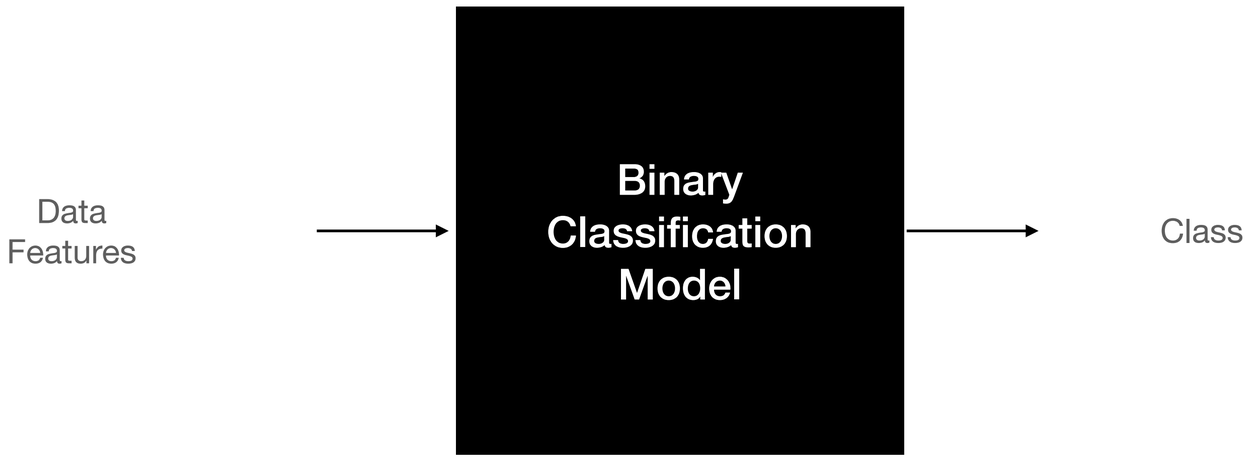


以人工的方式標注這些資料，
首先更新機率表，其次為每一群打分數

Step 4
Re-train model 更新模型權重



Step 1
Initial Model



Step 2
Select Data

作法

1. PCA + Cluster 分成一些群
2. 先挑選一些群
3. 從每一群中挑選 10 筆資料

舉例來說：
目前我們根據機率表選到 group2, group4, group6, group7。
我們會在這四群裡面各選10筆具有重要性的資料出來。

2 : 用 Exploration-Exploitation 決定要挑哪些群
3 : 用 Uncertainty Sampling 挑選資料

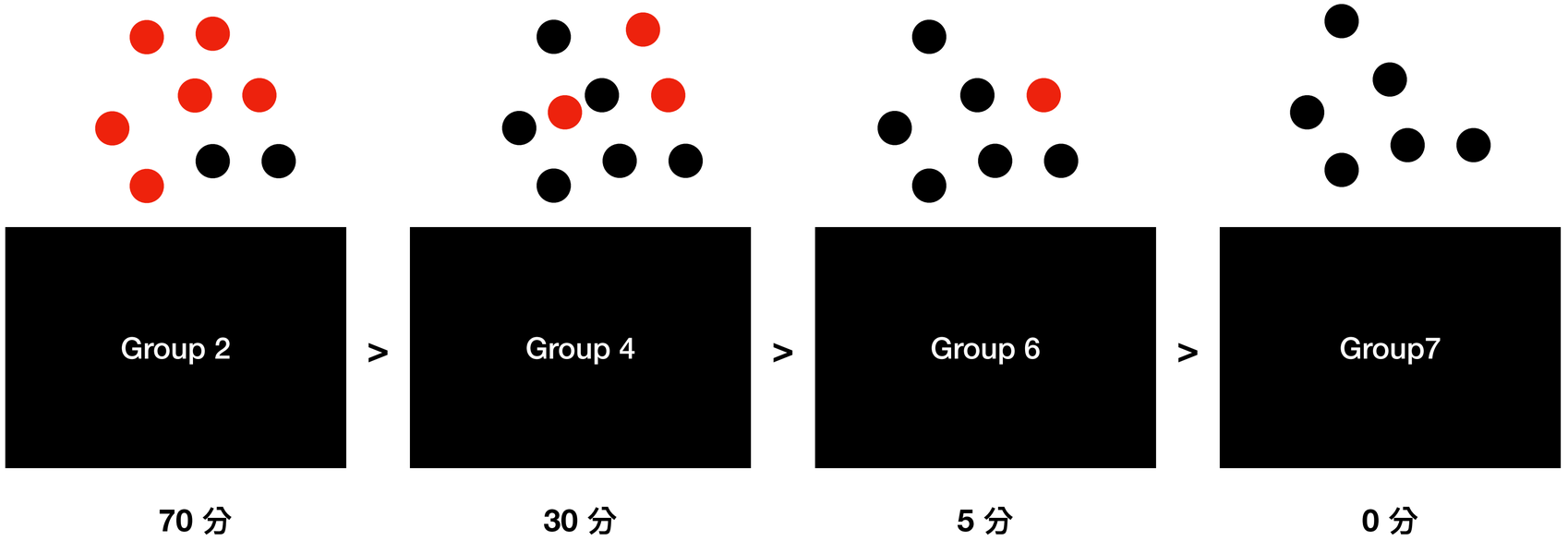
Exploration-Exploitation

1. 朴素bandit
2. Thompson sampling
3. Upper Confidence Bound
4. Epsilon-Greedy

Uncertainty Sampling

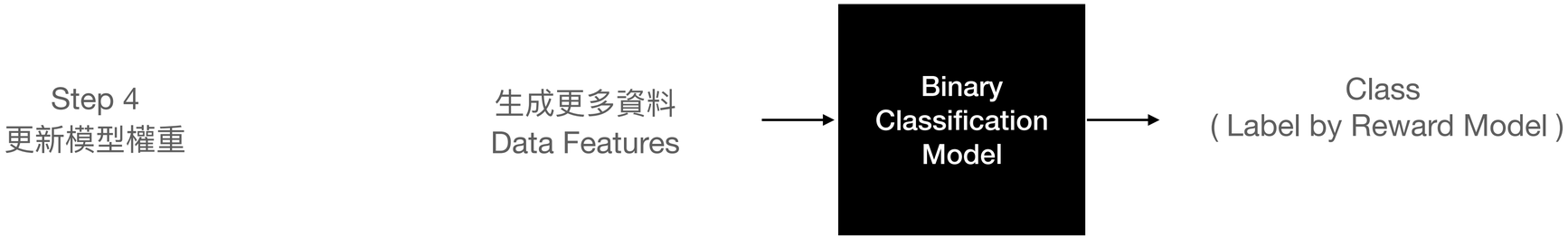
1. Random
2. Least Confident
3. Margin Sampling
4. Entropy

Step 3
Human Feedback



以人工的方式標注這些資料，首先更新機率表，其次為每一群打分數。

舉例來說：
把每一個 group 挑出來的資料當成是一個句子要打分數，裡面有幾個 Fraud data 就加幾分。
這些標注的資料會幫助我們更新機率表，並且用於訓練 Reward Model。
Generative Model 來作為 Reward Model。
將剛剛選到的其中 20 筆資料當作 Test ，來判斷何時停止訓練 Reward Model。

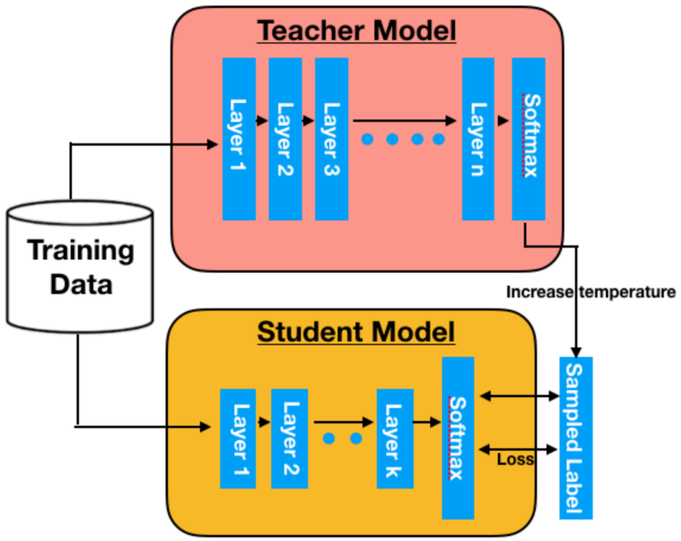


當 Reward Model 被訓練好，就可以去生成資料來訓練模型。

舉例來說：
假設標注到 1000筆時，發現Reward Model 效果不錯已經學會打分數，
這時就使用 Reward Model 對剩下的 Unlabeled 資料加上標籤。
把這些資料拿去給最終模型 Train ，查看效果。（由於這個資料集有所有資料的label）

Knowledge Distillation 知識蒸餾

考量到銀行可能需要較簡單的模型，因此最後可以把我們訓練的大模型轉換到小模型上。



知識蒸餾是抽取複雜模型訓練出的精華給小模型，讓這個小的簡單模型也能達到跟複雜模型一樣的效果。

其中 student 模型從可以更複雜的 teacher 模型中 「學習」 。如果已經透過複雜的結構建構出不錯的模型，可以用知識蒸餾訓練出較簡易版本的模型，準確度不會差太多。