Алгоритм Q-learning, применительно к задаче исследования случайного лабиринта

Студент: Калинин Владислав
Руководитель: Игорь Кураленок

Введение

- На сегодняшний день существует множество алгоритмов прохождения лабиринта

Введение

- На сегодняшний день существует множество алгоритмов прохождения лабиринта
- К таким алгоритмам относятся:
    * Алгоритм «Правой руки»
    * Алгоритм «Люка-Тремо»
    * «Волновой алгоритм» и т.д.

Введение

- На сегодняшний день существует множество алгоритмов прохождения лабиринта
- К таким алгоритмам относятся:
    - Алгоритм «Правой руки»
    - Алгоритм «Люка-Тремо»
    - «Волновой алгоритм» и т.д.

Постановка проблемы

- Дан случайным образом сгенерированный замкнутый конечный лабиринт и начальная позиция
- Изначально лабиринт не известен и познаётся по ходу продвижения по нему
- Задача научить машину изобретать алгоритм обхода лабиринта
- За основу взят метод Q-learning

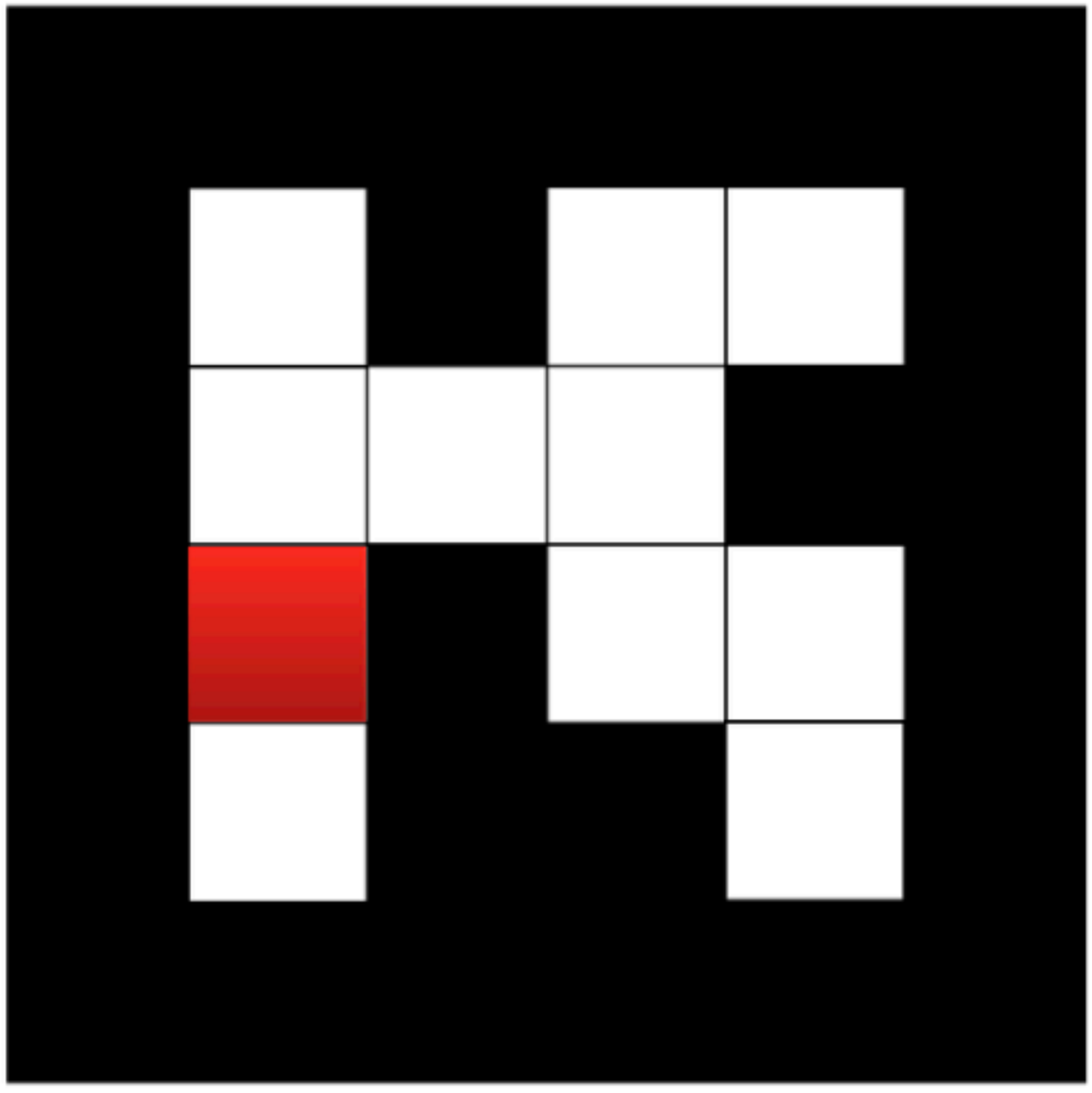
Q-learning

Игра агента со средой:

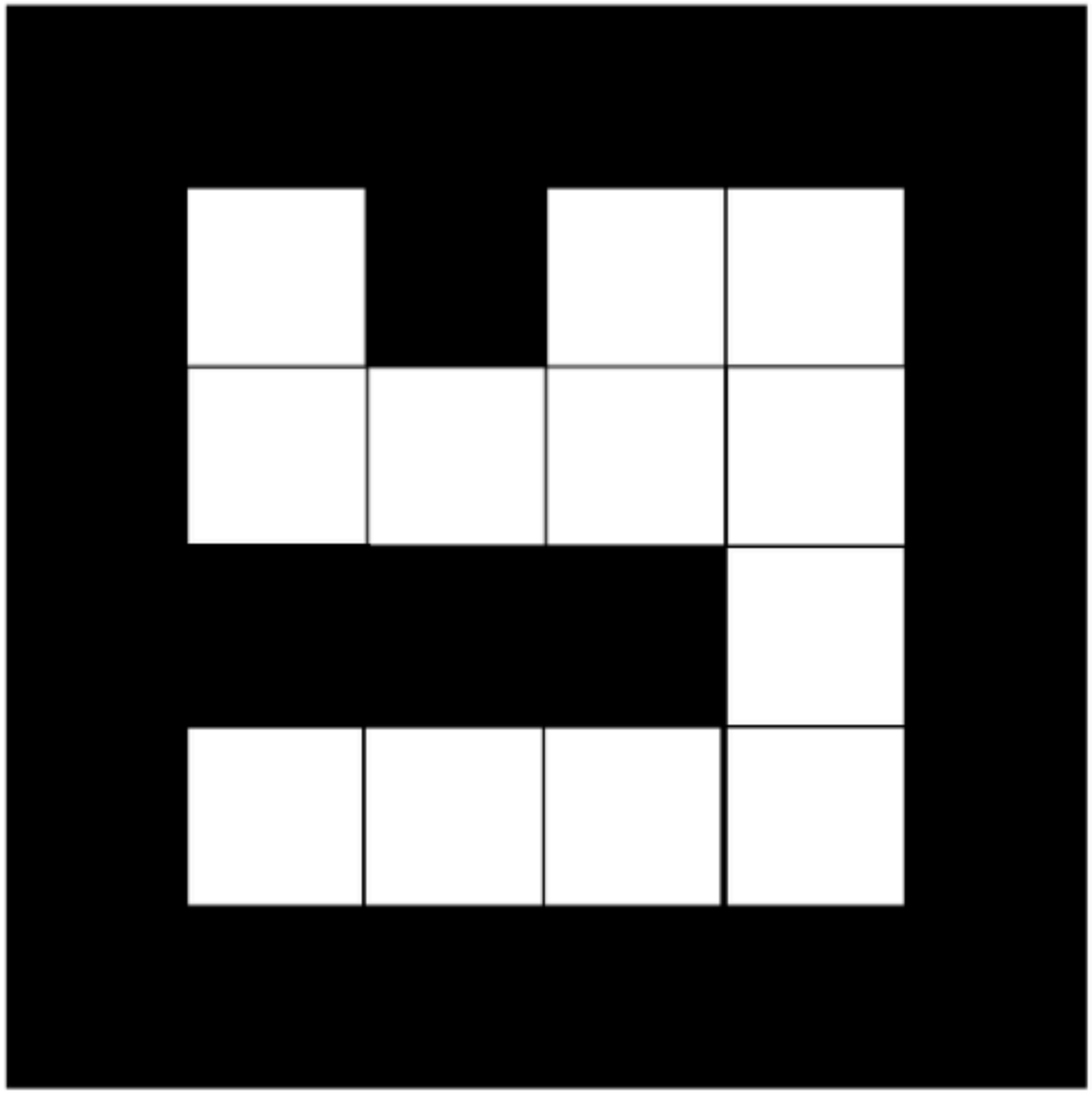
1: инициализация стратегии $\pi_1(a \mid s)$ и состояния среды s_1
2: **для всех** $t = 1, \ldots T, \ldots$
3: агент выбирает действие $a_t \sim \pi_t(a \mid s_t)$:
$a_t := \arg\max_a Q(s_t, a)$ — жадная стратегия;
4: среда генерирует $r_t \sim p(r \mid a_t, s_t)$ и $s_{t+1} \sim p(s \mid a_t, s_t)$;
5: $Q(s_t, a_t) := Q(s_t, a_t) + \alpha_t \big(r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)\big)$;

A — множество возможных *действий*
$p(r|a)$ — неизвестное распределение *премии* $r \in \mathbb{R}$ для $a \in A$
$\pi_t(a)$ — *стратегия* (policy) агента в момент t, распределение на A

Постоянный лабиринт



Случайный лабиринт



Первая попытка

- В качестве состояний взято:
    - Кратчайшее расстояние и направление до ранее неисследуемой позиции
    - Кратчайшее расстояние до стены

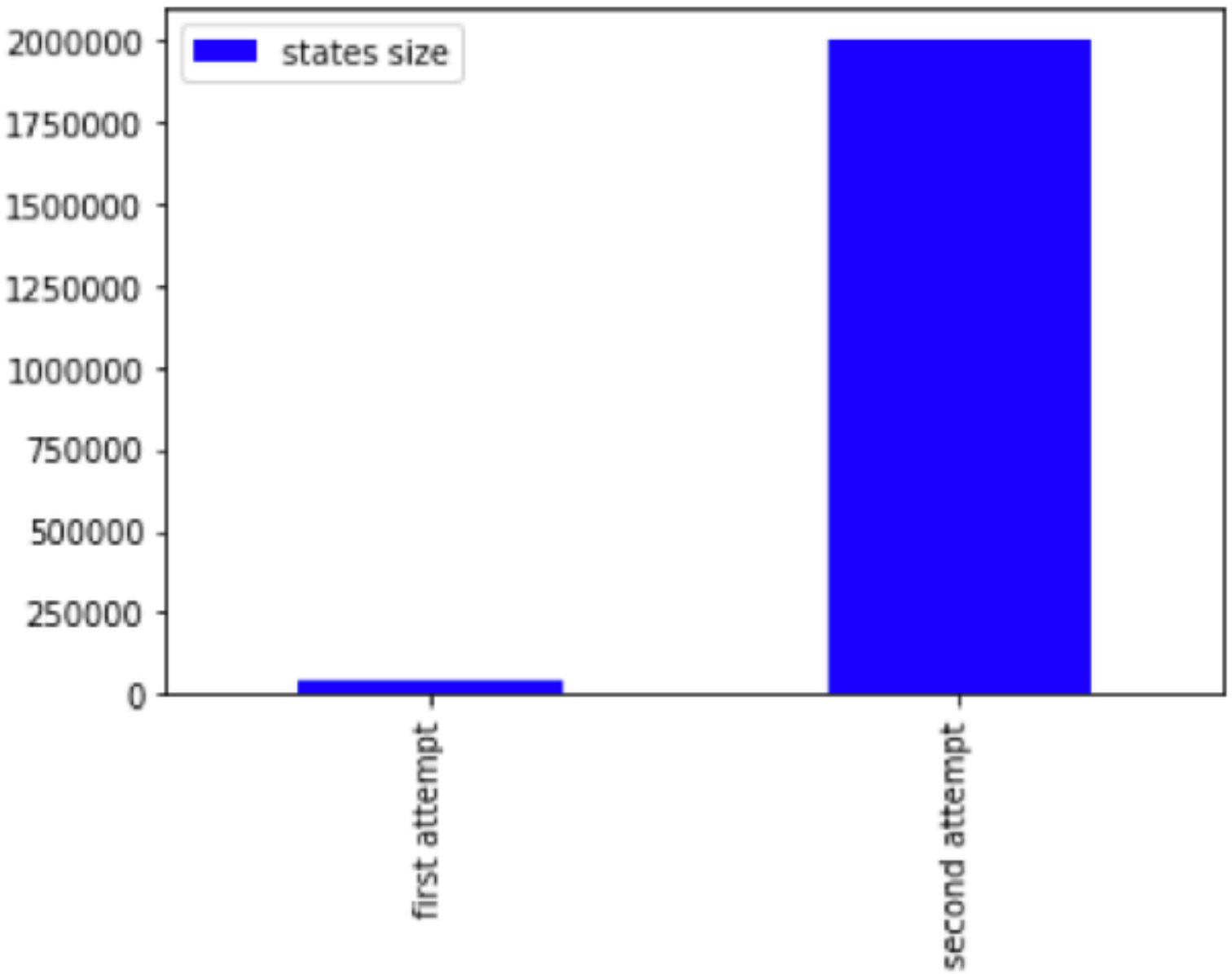
Первая попытка

- В качестве состояний взято:
    * Кратчайшее расстояние и направление до ранее неисследуемой позиции
    * Кратчайшее расстояние до стены

Вторая попытка

- В качестве состояний взято:
    * Кратчайшее расстояние и направление до ранее неисследуемой позиции
    * Расстояние до ближайшей стены в каждом из 4-ёх направлений (верх, низ, лево и право)
    * Типы тайлов на расстоянии 1 вокруг персонажа (стена, пол или неизвестный)

Комбинаторный взрыв



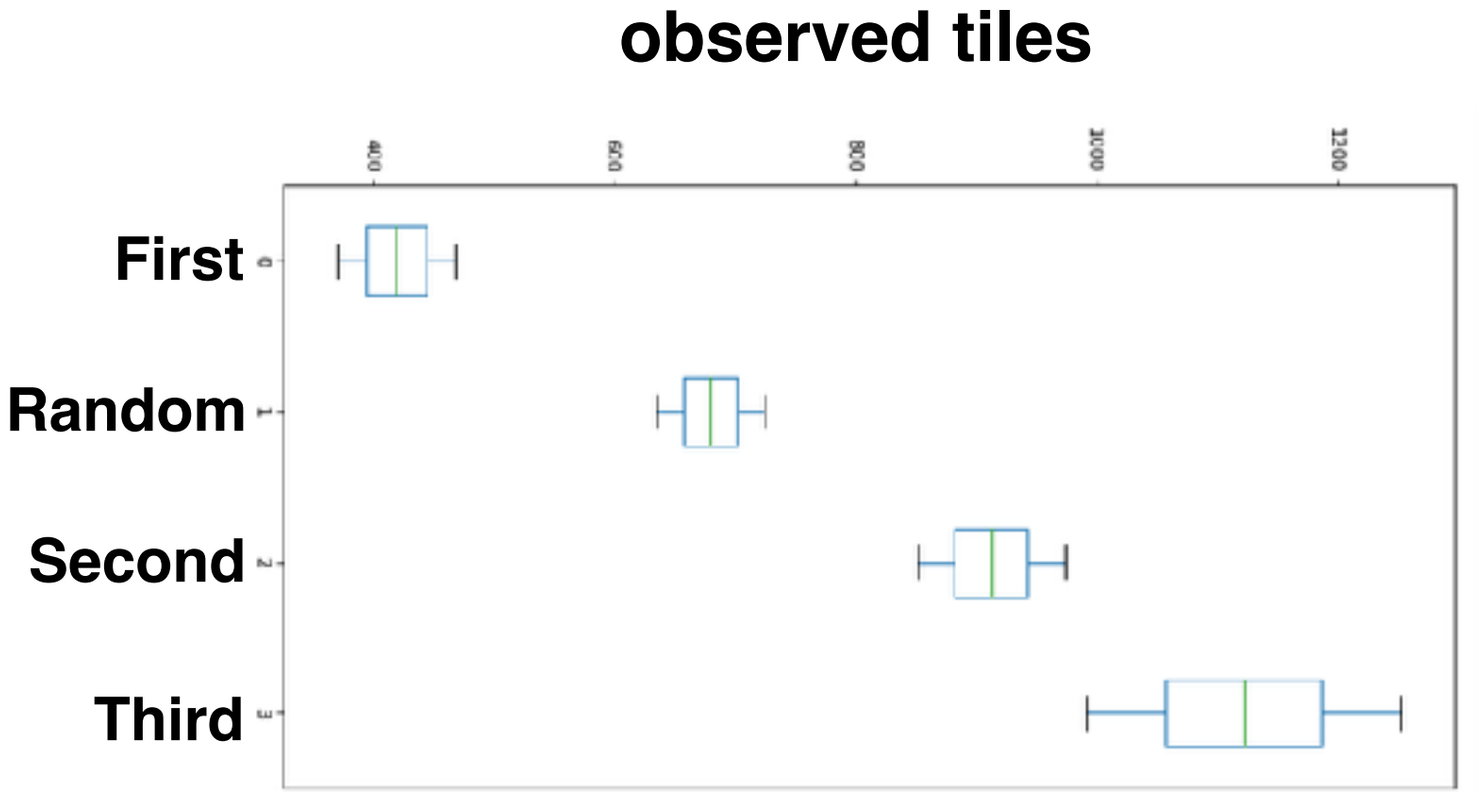
Третья попытка

- Факторизуем матрицу Q в виде функции двух векторов a - действия и s - состояния: $Q(a, s) = a^T A^T S s$, где A и S - матрицы
- В качестве условной вероятности берём softmax: $P(a|s) = \frac{e^{Q(a,s)}}{\sum_b e^{Q(b,s)}}$
- строим логарифмическую функцию правдоподобия:

$$L = \sum_{t=0}^{\infty} score \cdot log(P(a_t|s_t)) + (1 - score) \cdot log(1 - P(a_t|s_t))$$

- применяем метод максимального правдоподобия.
- Данный подход позволяет вводить сколь угодно большие множества состояний

Результаты


observed tiles
400
600
800
1000
1200
First
Random
Second
Third
0
1
2
3

Выводы

- Сделали ощутимый прогресс по сравнению со «случайным блужданием»
- Установили зависимость качества модели от числа характеристик карты

Дальнейшие исследования

- Использовать в качестве состояний не характеристики карты, а «саму карту»

Спасибо за внимание