

# Rozwiązywanie Sapera za pomocą sztucznej inteligencji

Dominika Zembrzуска, Kacper Ziółkowski, Mateusz Buchajewicz

04.05.2019

## 1 Wprowadzenie

Celem projektu jest utworzenie algorytmów opartych na sztucznej inteligencji rozwiązujących popularną grę Saper.

## 2 Zasady sapera

## 3 Wykorzystane technologie

## 4 Opis algorytmu

### 4.1 Bot generujący dane

Bot działa w oparciu o 3 główne metody rozwiązujące:

- Rozwiązywanie za pomocą prostej logiki
- Rozwiązywanie za pomocą układów równań
- Strzelanie w losowe pole

#### 4.1.1 Rozwiązywanie za pomocą prostej logiki

Sprawdza po kolei wszystkie pola planszy, zawierające wartość numeryczną. Jeśli liczba pól sąsiadujących ze sprawdzanym polem, a nieoznaczonych jako "bez miny" (czyli zarówno oznaczonych jako "miny", jak i pustych) jest taka jak wartość pola, to wszystkie te pola są oznaczane jako miny. Natomiast jeżeli liczba oznaczonych min na polach sąsiadujących jest równa wartości sprawdzanego pola, to wszystkie pozostałe pola sąsiadujące zostają oznaczone jako "bez miny"

#### 4.1.2 Rozwiązywanie za pomocą układów równań

Metoda opiera się na ułożeniu układu równań z pól na planszy oraz rozwiązaniu go. Algorytm przekształca układ równań na macierz  $n \times m$ , gdzie  $n$  jest liczbą oznaczonych pól z wyznaczoną wartością, a  $m$  jest liczbą różnych pól nieoznaczonych z nimi sąsiadujących.

Dla przykładu, poniższa plansza:



Rysunek 1: Przykładowa plansza sapera

Zostanie przekształcona do postaci:



Rysunek 2: Przykładowa plansza sapera z polami oznaczonymi jako zmienne

Otrzymany układ równań:

$$\begin{cases} A + B = 1 \\ A + B + C = 1 \\ D + E = 1 \\ E + F = 1 \\ A + B + C + F = 1 \end{cases}$$

Co z kolei można zapisać w postaci macierzy:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Macierz główna nie jest macierzą kwadratową (a przynajmniej nie zawsze), więc algorytm nie może określić dokładnych wartości wszystkich zmiennych. Może jednak przekształcić tę macierz za pomocą operacji elementarnych do takiej postaci, żeby mógł odczytać z niej jak najwięcej wartości zmiennych.

$$\left[ \begin{array}{cccccc|c} 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 \end{array} \right] \longrightarrow \left[ \begin{array}{cccccc|c} 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array} \right]$$

Co jest tożsame z układem równań:

$$\begin{cases} A + B = 1 \\ C = 0 \\ D = 0 \\ E = 1 \\ F = 0 \end{cases}$$

W tym przypadku doskonale widać, że pole E zawiera minę, natomiast pola C, D, F jej nie zawierają. O polach A i B nic nie wiadomo, co wynika z faktu iż macierz główna nie jest kwadratowa.

Warto jednak zauważyć, że operacje podstawowe nie zawsze zwrócą wartość 0 lub 1. W ogólności algorytm sprawdza otrzymane wyniki za pomocą prostych warunków:

$$\begin{cases} \text{if } \forall(a_i \in M_j^W) \mid \sum a_i = \sum |a_i| = |b_j| \text{ then } x_j = 1 \\ \text{if } \forall(a_i \in M_j^W) \mid \sum a_i = \sum |a_i| = 0 \text{ then } x_j = 0 \end{cases}$$

Analogicznie jak wyżej, wszystkie pola z wartością 1 algorytm oznacza jako pola "z miną", a te z wartością 0 jako "bez miny".

#### 4.1.3 Strzelanie w losowe pole

Ta metoda jest używana wtedy i tylko wtedy, gdy obie powyższe po przejrzaniu całej planszy nie dadzą rady oznaczyć ani jednego pola. Algorytm wybiera losowo jedno z nieoznaczonych pól i oznacza je jako pole "bez miny". Jeśli oznaczy poprawnie gra jest kontynuowana z wykorzystaniem powyższych metod. Jeśli pole nie zostało poprawnie oznaczone, gra jest zakończona porażką.

## 4.2 Sieć neuronowa

## 4.3 Prolog

## 5 Porównanie metod

Bot osiąga około 85-90% skuteczności na małych planszach (10x10, 10 min) oraz około 10-15% na dużych planszach (30x15, 99 min).

Algorytm	Mała	Duża
Bot	85-90	10-15
Sieć neuronowa	?	?
Logika / prolog	?	?

## 6 Wnioski