게임과 AI

선문대학교

글로벌소프트웨어학과

2017315055

에다카츠토시

목차

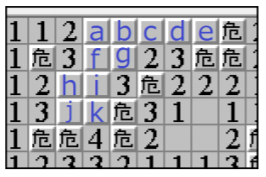
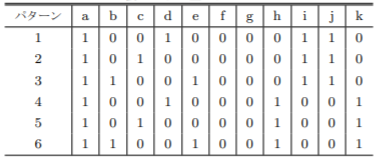
1. 요약 ……………………3
2. 연구 배경 및 동기 ……………………3
3. 선행연구 조사 ……………………4
4. 문제 정의 및 목표 ……………………6
5. 연구 방법 ……………………7
6. 주요 연구 결과 ……………………9
7. 결론 및 향후연구계획 ……………………10
8. 참고문헌 ……………………11
9. 요약

지뢰찾기는 1인용 퍼즐게임으로 플레이어가 모든 정보를 얻을 수 있는 것은 아닌 불완전 정보 게임이다. 즉 지뢰가 아닌 Block을 알 수 있는 안전한 방법이 존재하는 것은 아니며 승리가 되느냐의 여부는 운에 달려있다. 규칙이 단순하며 많은 사람들에게 친숙한 반면 최적의 전략 및 달성 가능한 승률의 상한은 아직 알 수 없다. 이는 지금까지 지뢰찾기AI를 사용해도 해결하지 못했던 문제이다. 본 연구에서는 16×16 Block에 지뢰가 40개의 지뢰찾기 게임을 지뢰찾기AI에 적용하여 정확도가 68.3%의 결과를 얻었다. 그리고 Original버전과 다르게 5번에 한번씩 지뢰위치를 바꾸는 Random버전도 작성하여 지뢰찾기AI에 적용한 결과 정확도가 92.6%이었다.

1. 연구 배경 및 동기

기존의 지뢰찾기AI는 문제를 해결하는데 있어 도중에 멈추거나 정확도가 낮은 수준으로 되어 있다는 문제가 있어서 지뢰찾기AI의 질을 높이고자 연구를 시작하게 되었다. 또한 지뢰찾기AI는 이미 있는 기술이기 때문에 본연구에서는 지뢰찾기를 Original버전과 5번에 한번씩 지뢰위치를 바꾸는 Random버전으로 2가지 버전을 만들고 지뢰찾기AI가 푼 결과를 비교하고자 하였다.

1. 선행연구 조사
2. Minesweeper Solver with Complete Enumeration of All Possible Patterns[[1]](#footnote-1)

일본 와세다대학에서 발표된 이 논문에서는 지뢰찾기의 안전한 Block을 판단하는 AI를 연구하였다.

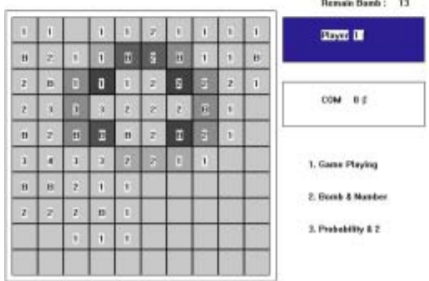
<표 1 패턴 표>

<그림 1 패턴 예측 그림>

이 논문의 지뢰찾기AI가 안전한 Block을 판단하는 방법은 먼저 Block이 지뢰인 것을 확정하면 위험으로 지정하고 지뢰 Block외 Block의 지뢰 패턴을 예측하고 가장 지뢰일 확률이 낮은 Block을 선택하는 방법이다.

결과로 이논문의 지뢰찾기AI는 정확도가 49.4%이었다.

1. 영향력 분포도 기법을 이용한 지뢰찾기 대전 AI[[2]](#footnote-2)

이 논문에서는 Block의 지뢰확률을 적합도로 표현하는 연구를 소개한다.

<표 2 영향력분포도 수치에 따른 표현색>

<그림 2 최종 적합도 영향력분포도>

이 논문의 지뢰찾기AI는 영향력분포도 기법을 사용한다. 주변의 열린 Block과 정보, 인근 지뢰의 개수를 데이터로 하여 각 Block의 초기 가중치를 계산하고, 주변 Block의 초기 가중치들을 합산해 최종 가중치를 얻어내는 방식이다.

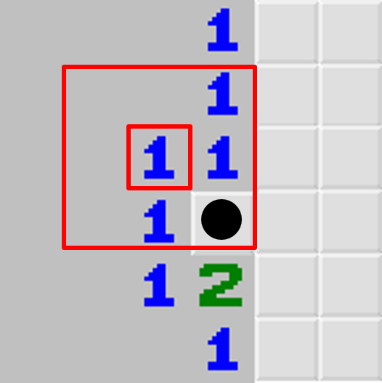
한계점으로 알고리즘 연산을 통해 초기 확률이 0이 된 경우와 아직 열리지 않아 0인 경우를 구분하지 못했다는 점이 있다.

1. 문제 정의 및 목표

* 문제 정의
* 기존의 지뢰찾기 AI는 자동으로 지뢰를 찾지만 도중에 지뢰가 어디 있는지 불확실한 경우에는 작동을 멈춘다.
* 선행연구에서의 지뢰찾기AI는 정확도가 49.4%으로 낮았다.
* 목표
* 마지막까지 풀어주는 AI를 제공한다.
* 지뢰찾기AI의 정확도를 향상시킨다.
* Original버전과 Random버전의 게임을 AI가 푼 결과를 비교한다.

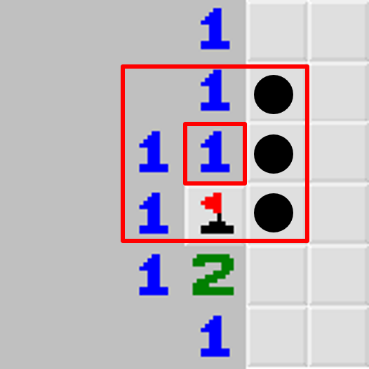
1. 연구 방법

문제 해결 방법은 아래와 같은 4가지이다.

1. 숫자 주변에 숨겨져 있는 Block 갯수가 숫자와 같으면 주변 Block에 Flag을 놓는다.

<그림 3 문제 해결 방법1의 예시>

열린 Block을 하나씩 보고 Block의 숫자와 주변에 열리지 않은 Block의 개수가 같을 시 Flag을 놓는다.

1. 숫자 주변에 Flag 갯수가 숫자와 같으면 주변 Block이 지뢰가 아닌 것을 기억한다.

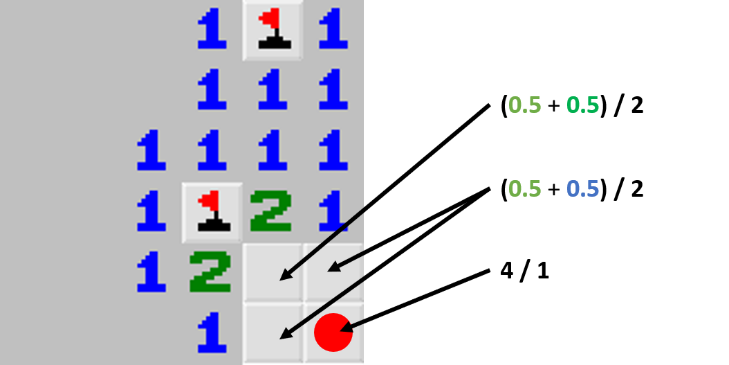
<그림 4 문제 해결 방법2의 예시>

열린 Block을 하나씩 보고 Block의 숫자와 주변에 Flag의 개수가 같을 시 주변의 열리지 않은 Block을 안전한 Block으로 인식하여 기억한다.

1. 위 방법을 적용하지 못하는 경우 지뢰가 아닌 Block을 Click 한다.

<그림 5 문제 해결 방법3의 예시>

1번 2번의 방법을 적용하지 못하는 경우 2번에서 기억했던 안전한 Block을 선택하여 열린다.

1. 위 방법을 적용하지 못하는 경우 지뢰확률을 계산하여 확률이 낮은 Block을 Click 한다.

<그림 6 문제 해결 방법4의 예시>

1번부터 3번까지의 방법을 적용하지 못하는 경우 모든 미확인 Block에 모든 미확인 Block의 개수를 나머지 Flag 의 개수로 나뉜 수를 기억한 후 열린 Block의 주변 미확인 Block의 개수를 Block의 숫자 빼기 주변 Flag개수로 나뉜 수의 평균을 계산하여 기억하고 제일 지뢰확률이 낮은 Block을 선택하여 열린다.

위 단계를 방복하여 문제를 해결한다.

1. 주요 연구 결과

실험결과는 아래와 같다. (실제 적용 결과는 참고문헌의 시연동영상URL을 참조하기 바람)

|  |  |  |
| --- | --- | --- |
| Version | Original | Random |
| 실행횟수 | 1000회 | 1000회 |
| 정확도 | **68.3**% | **92.6**% |
| 속도 최대값(ms) | 2691 | 1541 |
| 속도 최소값(ms) | 1150 | 614 |
| 속도 평균값(ms) | 1894 | 822 |
| 표준편차 | 428.1 | 204.5 |

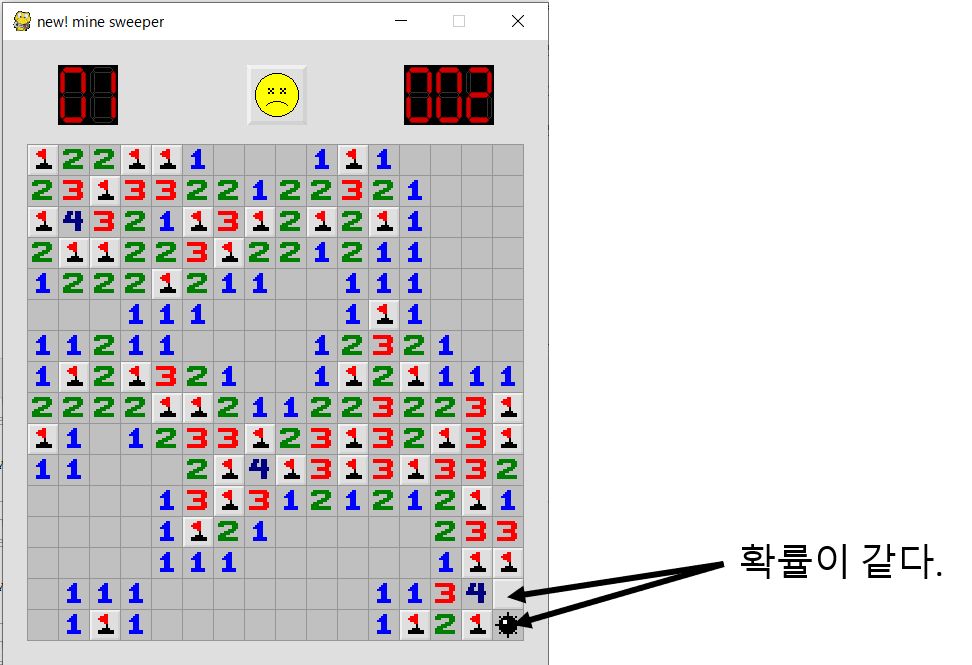
<표 3 Original버전(좌)와 Random버전(우)의 실험결과>

실행을 Original버전과 Random버전을 각 1000회씩을 한 결과 Original버전에서는 정확도가 68.3%로 기존 연구보다 높은 수준을 얻을 수 있었다. 속도는 많아도 3초 있으면 해결할 수 있었다.

본 연구에서는 지뢰게임Random버전과 Original버전을 비교하였다. Original버전 보다 Random버전이 풀기 어렵다고 예측하였지만 모든 면에서 AI에게는 Random버전이 해결하는데 있어 우수한 것을 알 수 있다.

1. 결론 및 향후연구계획

지뢰찾기AI는 Random으로 지뢰 위치를 바꿀 시 새롭게 지뢰 위치를 예측할 기회가 생기고 그 전에 지뢰 위치를 예측하지 못해도 지뢰인 확률이 있는 Block을 선택할 필요가 없어지기 때문에 수준 높은 결과를 얻을 수 있었다.

본 연구의 지뢰 찾기AI는 정확한 지뢰 확률을 계산하여 진행하기 때문에 정확도 68.3%를 얻을 수 있었다.

<그림 7 실패사래>

하지만 그림7 처럼 미확인 Block의 지뢰확률이 같을 경우가 가끔 생기기 때문에 더 이상 정확도를 향상시킬 수 없었다. 그리고 이번에는 16×16 Block에 지뢰가 40개의 지뢰찾기 게임만 지뢰찾기AI에 적용하였기 때문에 지뢰찾기AI 정확도를 말하기에는 설득력이 부족하다는 한계점이 있다.

따라서 향후 계확으로는 다양한 Block의 그기와 지뢰의 개수를 설정할 수 있는 환경에서 실행하여 조건에 따라 알고리즘에 부족함이 있는지를 확인할 예정이다.

1. 참고문헌

[1] Eichi Omori, Masato Inoue (2012) "Minesweeper Solver with Complete Enumeration of All Possible Patterns“, 研究報告ゲーム情報学（GI） 2012-GI-28(5), 1-6

[2] Sunhwa Lee, Kyungeun Cho (2010) "AI for Minesweeper Game using Influence Map Method“, KOCCA 2010-4

[3] <https://uesyuu.github.io/minesweeper/minesweeper.html>

[4] <http://www.nuis.ac.jp/~nakada/researches/Minesweeper/>

[5] 시연동영상 URL

https://youtu.be/lrOSd4aACrg

1. Eichi Omori, Masato Inoue (2012) "Minesweeper Solver with Complete Enumeration of All Possible Patterns", 研究報告ゲーム情報学（GI） 2012-GI-28(5), 1-6 [↑](#footnote-ref-1)
2. Sunhwa Lee, Kyungeun Cho (2010) "AI for Minesweeper Game using Influence Map Method“, KOCCA 2010-4 [↑](#footnote-ref-2)