

Training a computer to learn Tic Tac Toe

Introduction

Machine Learning is the subfield of computer science that gives computers the ability to learn without being explicitly programmed. In this project we use tree data structure to train the computer to learn the famous game of tic tac toe.

Algorithm

Constructing the tree- A tree of 8 levels is made such that each node has 9 children representing each 1X1 square in 3X3 square of the tic tac toe. Each node has links to all his children along with three integer variables, win(no. Of times player-1 won), loss(no. Of times player-2 won), draw(no. Of times match was a draw).

Training the tree- Random set of legal moves were generated and was use to fit the tree. For Example-
If the 1X1 squares of tic tac toe is indexed in the following way-

| | | |
|---|---|---|
| 0 | 1 | 2 |
| 3 | 4 | 5 |
| 6 | 7 | 8 |

The set of moves- 0, 4, 3, 2, 6 will result in win of player-1.
(Green color number represents the moves played by player-1 and blue by player-2) was used to increment the value of win (int variable) .

At depth 1, node 0->win++
At depth 2, node 4->win++
and so on...

Alternatively when player 2 won, loss (int variable) was incremented and draw(int variable), when match resulted in a draw.

Predicting the next move-

The next move was predicted using a formula-
 $\text{confidence} = x * \text{win} + y * \text{loss} + z * \text{draw}$

- Computer as player 1- $x=2, y=-2, z=1$.
- Computer as player 2- $x=-3, y=2, z=2$.

Higher the confidence more probability of playing that move.

Note: The values of x, y and z are subject to question and its value is completely based on intuition and few tests. Although a better way will be to analyze test data and use a minimizing function (like gradient descent) to determine the values.

Performance Issues-

This program is computationally very costly both in terms of memory and time. The computer may take upto few minutes to construct and fit the tree.

What to do when the program doesn't work due to overflow ?

Use the file **TryThisIfOtherDidNotWork.cpp** (read note) or Alternatively, **modify- AI in tic tac toe.cpp** as following:

- In function `makethetree(node*, int)` : Line no. 38:
 - change `level==9` to `level==8`
- In `main()` : Line no. 316, 317:

- Change if(size>8)
size=8
to if(size>7)
size=7

Note: The above change or usage of other file will result the program to play only upto 7 moves in total, but user will still be able to play the game.

Screenshots

(below)

```
Terminal Shell Edit View Window Help
Nikhil Final — a.out — 181x49

Player 2 won
Press 0 to exit and any other number to play again
1
Press
1 to be player-1 (x)
2 to be player-2 (o)
1
|0||1||2|
|3||4||5|
|6||7||8|
Enter the next move
0
|x||1||2|
|3||4||5|
|6||7||8|

Computer played: 4
|x||1||2|
|3||o||5|
|6||7||8|
Enter the next move
8
|x||1||2|
|3||o||5|
|6||7||x|

Computer played: 2
|x||1||o|
|3||o||5|
|6||7||x|
Enter the next move
6
|x||1||o|
|3||o||5|
|x||7||x|

Computer played: 3
|x||1||o|
|o||o||5|
|x||7||x|
Enter the next move
7
|x||1||o|
|o||o||5|
|x||x||x|
Player 1 won

Press 0 to exit and any other number to play again
```

```
Terminal Shell Edit View Window Help
Nikhil Final — a.out — 181x49

Hey, I am Nikhil :) .Lets play tic tac toe!!
This program lets computer train itself to win the game!!
Press any button to continue...
1
[Status]: Constructing the tree...
Enter the number of times you want to train the tree with random sequence of moves
Recommended times: 10^6 (usually takes upto 4 mins to train)
1000000000
This might take few minutes
[Status]: Fitting the tree...
Press
1 to be player-1 (x)
2 to be player-2 (o)
1
|0||1||2|
|3||4||5|
|6||7||8|
Enter the next move
4
|0||1||2|
|3||x||5|
|6||7||8|

Computer played: 6
|0||1||2|
|3||x||5|
|o||7||8|
Enter the next move
2
|0||1||x|
|3||x||5|
|o||7||8|

Computer played: 8
|0||1||x|
|3||x||5|
|o||7||o|
Enter the next move
0
|x||1||x|
|3||x||5|
|o||7||o|

Computer played: 7
|x||1||x|
|3||x||5|
|o||o||o|
Player 2 won
Press 0 to exit and any other number to play again
Platform and Plugin Updates: IntelliJ IDEA is ready to update. (5 minutes ago)
```