

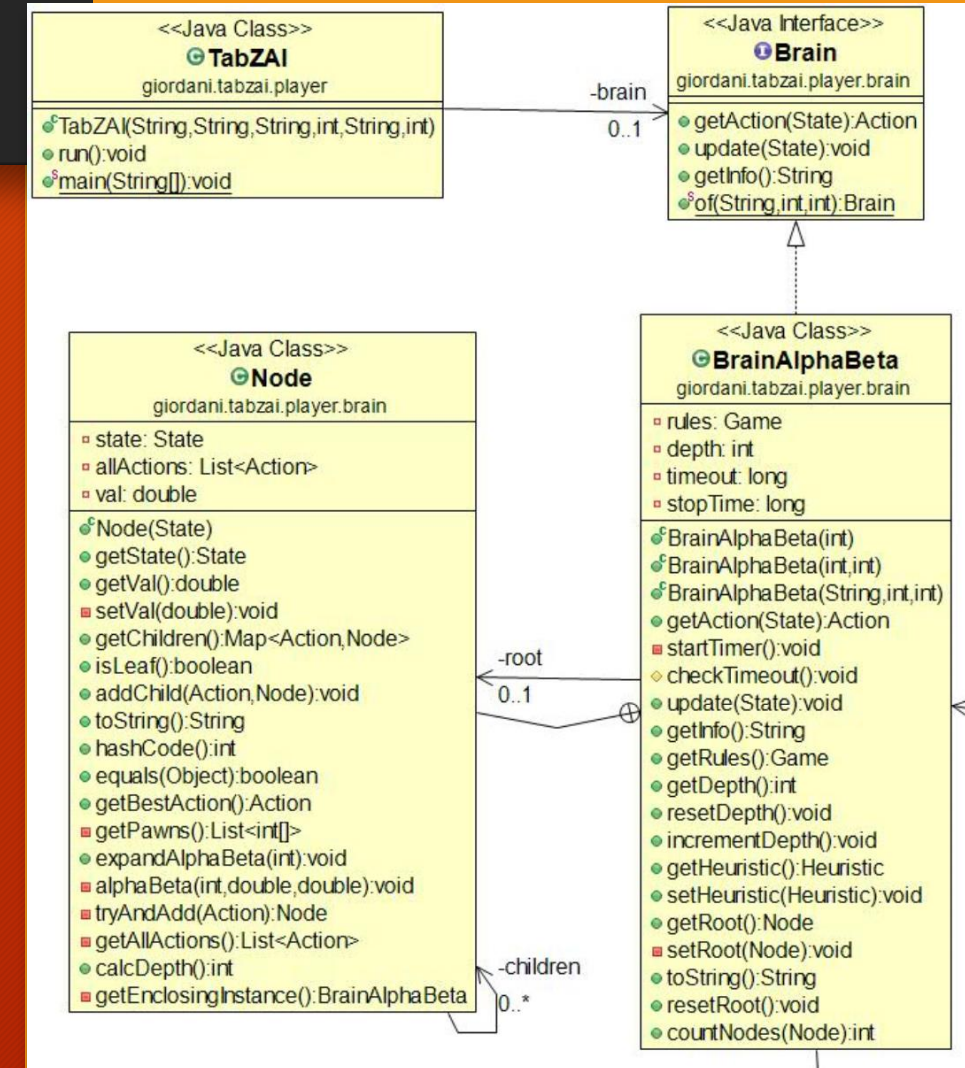
Simone Giordani

Fundamentals of Artificial Intelligence - Unibo

Tablut Challenge 2020 TabZAI

Alpha-Beta Pruning

- The best move is selected using tree search with Alpha-Beta pruning
- The maximum depth of the tree is increased at runtime
- The timeout check is done at each node's expansion in order to interrupt the search if the computation time reach 90% of the maximum time limit
- The remaining time is used for the backtracking of the last computed values and for the final move selection
- Terminal states evaluation:
 - White win: +infinity
 - Black win: -infinity
 - Draw: 0

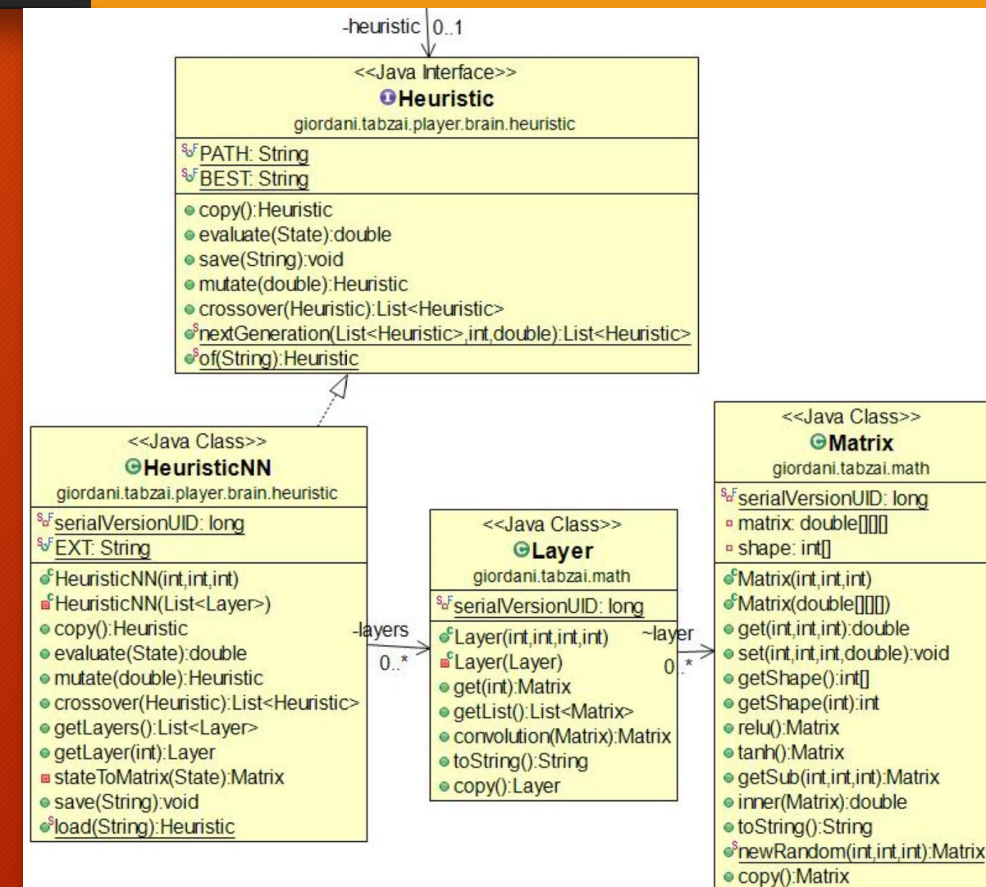


Heuristic

- Inspired to Convolutional Neural Networks
- Input shape: (9, 9, 1)
 - The white pawns are substituted with the value 1
 - The black pawns with the value -1
 - The king with the value 10
 - The empty cells with the value 0
- 4 layers with shapes:

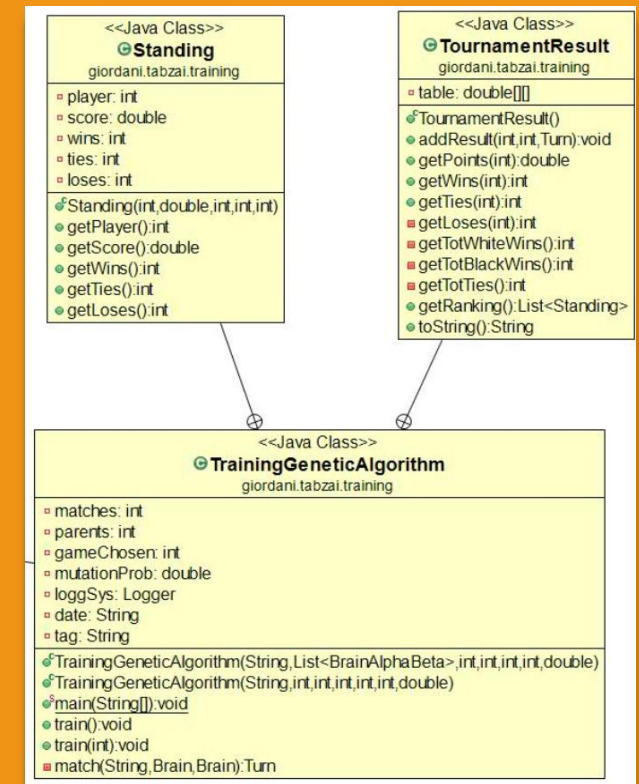
1. (64, 3, 3, 1)	Input layer
2. (32, 3, 3, 64)	Hidden layer 1
3. (16, 3, 3, 32)	Hidden layer 2
4. (1, 3, 3, 16)	Output layer

 - Convolution done with stride 1 and no padding
 - Using tanh() as activation function
 - Total number of parameters: 23,760
- Output shape: (1, 1, 1)
 - The only element of this matrix is the real number representing the evaluation of the State



Training

- Trained with a Genetic Algorithm
- Population of 8 players evaluated through an all-play-all tournament
- The 2 best players will be the parents of the next generation
- The new generation contains the 2 parents, the 2 children obtained through the crossover of the layers, plus 4 mutated child
- The training evaluated 100 generations, for a total of 5,600 simulated games
- With a timeout of 2 seconds for each move selection, the training lasts 4d:13h:26m



Results of the 100° tournament:

Match results (1=white, 0=black, 0.5=draw)

	0	1	2	3	4	5	6	7
0		0.5	0.5	0.5	0.5	0.5	0.5	1
1	0.5		0.5	0.5	0.5	0.5	0.5	0.5
2	0.5	0.5		0.5	0.5	0.5	1	0.5
3	0.5	0.5	0.5		0.5	0.5	0.5	0.5
4	0.5	0.5	0.5	0.5		0.5	0.5	0.5
5	0.5	0.5	0.5	0.5	0.5		0.5	0.5
6	0.5	0.5	0.5	0.5	0.5	0.5		0.5
7	0.5	0.5	0.5	0	0.5	0.5	0.5	

Ranking

P	S	W	T	L
3	7.5	1	13	0
2	7.5	1	13	0
0	7.5	1	13	0
5	7.0	0	14	0
4	7.0	0	14	0
1	7.0	0	14	0
6	6.5	0	13	1
7	6.0	0	12	2

Training

- The weight of the layers are initialized with a Gaussian distribution
- The mutation is done by adding a random number, obtained by a Gaussian distribution, to each parameter of the layers, with a probability of 50%
- In the figure there is a representation of the 64 filters of the first layer of the last neural network
 - Green = positive parameter
 - Red = negative parameter
- These filters are the patterns that the network looks around the board to evaluate the position

