Give Your Quagent A Neural Brain

Programming Assignment #7
CSC 481 – Spring '10

This is **NOT** a Team Assignment

Problem Statement

Building on the results of Programming Assignment #6 ("Train your Quagent") we ask the question whether artificial neural networks can be used to build a model of walking through a room for the Quake 2 quagents. As in Assignment #6, the task at hand is to walk along the walls of the Empty Room in a counter clockwise fashion (it would be ok to cross the room the first time from the spawn point). There is no stopping criterion, the quagent simply keeps on walking along the walls until it dies of old age. The goal is to have a neural network in control of the navigation. A good place to start is the training dataset (table) you developed for the ID3 tree builder. Convert this table into a CSV ("comma separated values") file using 0 for (lrf)clear and 1 for (lrf)blocked. In the target attribute use 0 to represent left, 1 to represent right, and 2 to represent walk. Here is a small snippet of an example CSV file:

```
Left, Right, Front, Navigate
0, 0, 0, 2
0, 0, 1, 0
1, 0, 1, 1
```

IMPORTANT: you must conform to these conventions, otherwise the learner framework will not work. Now build an appropriate neural network with the Tiberius software (available from the course website) and generate Java code for this neural network. Copy the generated Java code and the neural network adapter (available from the course website) into the folder where you saved the learner framework. Compile all the Java files and run your quagent program. **Do not modify the generated code!** (there is one exception…you will need to comment out the warning statement at the beginning of the Java code generated by Tiberius for the neural network)

Experiments:

(1) Was your ID3 training dataset enough or did you have to add more observations to the table to properly train a neural network?
(2) What is the minimum number of observations in the CSV file that will result in a properly trained neural network?
(3) Can a neural network with a single hidden non-linear node learn the navigation table?
(4) What is the smallest neural network that can learn your CSV training data?
(5) What is the minimum number of epochs required to train the smallest network?

Deliverables

The CSV file with the training instances, the generated neural network Java code, and a brief writeup answering the questions above.

Submitting your Project

Submit your work by email to hamel@cs.uri.edu by **Monday April 5th 10pm**.