CSE/ECE 848 Introduction to Evolutionary Computation

Module 3 - Lecture 15 - Part 4

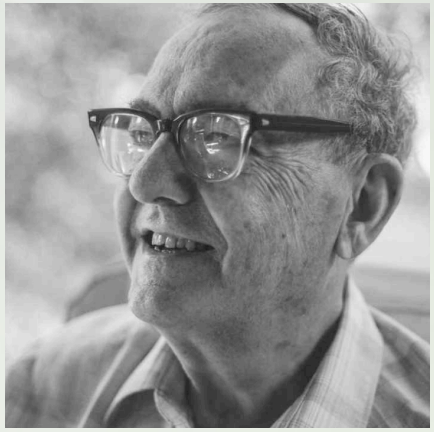
Learning Classifier Systems Rule-based Systems

Wolfgang Banzhaf, CSE
John R. Koza Chair in Genetic Programming

Early AI

… was focused on general purpose problem solvers, what were eventually called “weak” problem solvers

- General Problem Solver (GPS)
- Worked by decomposing a problem into solvable parts, then reassembling the solution
- Herbert Simon, late 50’s, early 60’s



Knowledge Based Systems

Reaction to that was to make systems that were very smart at solving particular, problems, so called “strong methods”

- Stanford Heuristic Programming Project
- Dendral, Mycin
- Edward Feigenbaum, 60’s through 80’s

Rules or Classifiers

- General approach was called a “production system”
- A production system was basically a rule-based system with rules of the following form:
- if <condition> then <action>

Actions

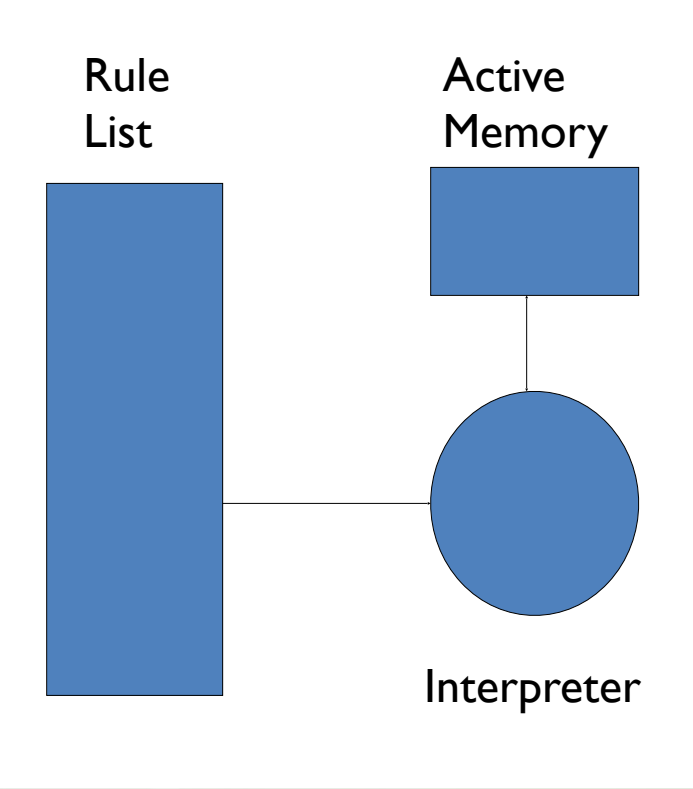
- Could post something (a token) to working memory
- Would generally output something to the world
- Could commit a change to working memory (update an existing value in some way)

Working memory was the site of interaction of rules

Typical Rule-Based System (RBS)

The cycle:

- Find first rule that applies
- Fire that rule, update memory
- Repeat until done (no more rules applies)



Post production systems

Post canonical system, Emil Post (1940s).

"A string-manipulation system that starts with finitely-many strings and repeatedly transforms them by applying a finite set of specified rules of a certain form, thus generating a formal language. Today they are mainly of historical relevance because every Post canonical system can be reduced to a string rewriting system (semi-Thue system), which is a simpler formulation. Both formalisms are Turing complete."

- Production systems were "easy" to program and amenable to "knowledge acquisition". The labs at Stanford and other places used them for the first "expert systems".

Advantages

The advantage of such rules were:

- They captured the intuition that knowledge was realized in little independent chunks
- That different reasoning techniques could use the same knowledge
- That knowledge would be easier to maintain

Disadvantages

- The rules, to be effective, had to "chain together" to solve a problem. Their independent nature made that difficult at times
- The independence of the problem-solver proved difficult to implement on "real" rule systems
- Classifier systems would address this

Holland and GAs

- Holland was interested in designing a way to use EC type approaches to program rule-based systems
- This would be important, because "programming" is difficult in this environment
- Holland pioneered classifier systems as his approach to this problem

Classifier System

Consists of three basic parts:

- A “rule” and “message” system
- A system for the apportionment of credit
- A genetic algorithm (of course)

Differences to other RBSs

- Classifier does parallel rule activation
  - Every rule that can fire does fire
- Rule rating not fixed by the programmer but based on performance metrics
- Rules are written in a way that they can be machine modified by a learning procedure