Artificial Intelligence Workshop

Course Structure

This course consists of five projects in artificial intelligence, which collectively address many of the major areas of A.I., including low and high-level perception, natural language, knowledge-management, and the modeling of rational thought, creativity, and learning; and involve several of its famous methods, such as genetic algorithms, neural networks, heuristics, semantic networks, and statistical algorithms.

These five projects will take about two weeks each, leaving the last three weeks for individual A.I. projects. The instructor will also be soliciting students to help on projects of his own.

The course projects will be programmed in Java. Parts of each project will be provided for students to extend and interface with. While we will discuss the principles behind some efficient algorithms, we will not emphasize efficiency.

Grading will be time-based. Students will keep track of their time on readings and exercises, and their grades will be determined by the fraction of the required time that they complete. Additional readings and assignment extensions will be available for those who are interested.

Class meetings will follow the standard schedule for a 2 unit laboratory course.

Reading: 1 hour / wk

About 20 pages per week from a variety of sources, describing problems and methods, and discussing related projects and philosophical issues.

Lecture: .5 hour / wk

Lectures will be used to introduce new methods and describe the projects. These may be grouped with the discussions in a two hour meeting every two weeks.

Discussion: .5 hour / wk

Unlike many areas of engineering, meaningful and interesting discussions of the mechanics involved in A.I. are readily available to introductory students. We will discuss students' own ideas for projects and methods, and their connections to other projects, and talk about the readings.

Programming: 4 hours / wk

There will be an extended class meeting every two weeks for working on projects. All project pieces can be done in groups or individually.

Additional assignments, readings, discussion topics, and projects may be incorporated as time permits, and are listed in another document.

1 Twenty Questions Project

1.1 Project Goal

Build a class-wide neural network to play the game “20 Questions”, inspired by `http://20q.net/`. However, the only answers that will be applicable are “abstracts”, and one goal of this project is to develop a schema for categorizing them.

1.2 Conceptual Setup

A neural network will grow with each new possible question and answer, following patterns to learn which questions are most useful to ask. Every question is an input node, which connects to every possible answer, which in turn connects to every question. Nodes will have fuzzy threshold; edge weights will be calculated by back propagation. Each student computer will maintain a piece of the network with a centralized server to facilitate communication. The network will start small and grow as additional questions and abstracts are added.

1.3 Assignments

- Build neural network nodes and edges classes, with communication between nodes through IP packets. [Function for synchronizing with the central server provided] (2 hr).
- Implement back propagation (2 hr).
- Extend the system (2 hr).

1.4 Readings

- Excerpt from *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches* by David White and Donald Sofge (*description of neural network method; 20 p.*)
- “The Evolution of Consciousness” by Daniel Dennet, from *Speculations*, ed. John Brockman (*computer-mind metaphors; 19 p.*)
- Excerpt from *Alan Turing: The Enigma* by Andrew Hodges (*computer as brain, development of Turing machine; 15 p.*)

2 Induction Collector

2.1 Project Goal

Use clustering methods and simulated annealing to combine web pages into abstractions that match each other both in content (using word frequencies) and "causation behavior" (using links).

2.2 Conceptual Setup

The human idea of causality is based on a complex abstraction process which selectively chooses some characteristics and correlations between instances for its conceptual boundaries while ignoring others. Consider the internet as a ready-made network of "instances", each "leading to" other instances. A simple clustering process would be insufficient for creating such abstractions, but we can develop a better model by combining a clustering framework with a energy-minimizing system (through the analogy of "understanding" as low-energy structure).

2.3 Assignments

- Create word-histogram, distance, and group-combining functions for clustering (web page extraction provided) (3 hr).
- Program the basic simulated annealing loop, change options, and energy-evaluation function. (3 hr)

2.4 Readings

- "Algorithm of the Gods" by Shawn Carlson, from *Scientific American*, March 1997 (*simulated annealing; 3 p.*)
- "Example-Based Learning for View-Based Human Face Detection" by Kah-Kay Sung and Tomaso Poggio (*Clustering and low-level perception; 13 p.*)
- Excerpt from *The View from Nowhere* by Thomas Nagle (*induction and knowledge; 15 p.*)

3 Genetic Sentence Structures

3.1 Project Goal

Evolve English sentences structures (phrases and sentence templates) that could be used for sentence creation. Learn about logical structures, inheritance, and genetic algorithms.

3.2 Conceptual Setup

A large body of text will serve as the "environment" in which virtual organisms will compete. The organisms' DNA will consist of structured templates, involving individual words and other organisms (the primitive instructions will include OR(x, y), SERIAL(x, y), and IS-A(x)). Each sentence will provide an opportunity for organisms to prove their usefulness at describing sentence structure.

3.3 Algorithm

1. Start at beginning of document with an empty pool of organisms.
2. Read a sentence; add its words to the pool of words
3. Create a new organism by combining words
4. Break the sentence down into a set of organisms and unclaimed words; in cases of ambiguity, have a metric for choosing the fittest.
5. For ever organism so used, improve its fitness and create an offspring by applying an additional combination, removing a combination, or a point mutation.
6. Continue from step 2. When too many organisms exist, randomly delete some based somewhat on fitness.

3.4 Assignments

- Design an logical instruction set and mutation routines (1 hr).
- Make algorithm for breaking sentence to various organisms (2 hr).
- Complete organism lifespan loop [Functions for parsing sentences provided] (1 hr).

3.5 Readings

- Excerpt from *Genetic Programming IV: Routine Human-Competitive Machine Intelligence* by John R. Koza, et. al. (*uses of G.A., basic form; 15 p.*)
- "Speculations of Reality: Deciding What to do Next" by William Calvin, from *Speculations* (*randomized language model; 20 p.*)
- Excerpt from *Patterns of the Hypnotic Techniques of Milton Erickson, M.D.* by Richard Bandler and John Grinder (*structure of thinking/language; causality; 10 p.*)

4 Interaction Modeling

4.1 Project Goal

Model the interactions of a number of independent entities through constraint propagation. Develop an elementary set of actions (e.g. `push`, `hold`, `break`) and relations (e.g. `left-of`, `within`, `rigidly-connected`) and use them to create composite behaviors and reactions for the entities. Learn about action specification, constraint propagation, and message-passing.

4.2 Conceptual Setup

Rule-based systems can develop unexpected, complicated behavior from simple rules. We will create a virtual world with a variety of entities, such as boxes, bananas, and professors. Each entity will have a set of rules, in addition to a global rule-set, which will form its behavior, and the entities will interact by sending messages (truth statements, like `push(me, left)`) to each other and responding to messages they receive. One message to which autonomous entities will respond is the `clock-tick` message; another message, recognized by all entities, will be `newrule(...)`, a message for giving entities to give each other new behaviors.

4.3 Assignments

- Complete a constraint propagation and message-passing system, partially provided (4 hr).
- Design a set of heuristics for the elementary actions and relations that we design in class (some rules provided) (2 hr).
- Complete one entity, and design two other entities (2 hr).

4.4 Readings

- Excerpt from *Computer Power and Human Reason: From Judgment to Calculation* by Joseph Weizenbaum (*early AI thoughts and Eliza; 15 p.*)
- "Enabling Agents to Work Together" by R. V. Guha and Douglas B. Lenat (*knowledge as independent agents; 16 p.*)
- Excerpt from *The Computer and the Mind: An Introduction to Cognitive Science* by P.N. Johnson-Laird (*A.I. learning; 15 p.*)

5 Image Creator

5.1 Project Goal

Create randomized images that emerge out of perceived structures grown on other perceived structures. More importantly, create a model of creative perception based on the CopyCat project (Mitchell and Hofstadter).

5.2 Conceptual Setup

Creativity is a chaotic process, and a model of it natural includes interaction between processes of insight and molding. The CopyCat project, a program for solving letter-analogies through high-level perception, uses a powerful model for combining these processes. This framework consists of atomic processes, which together peruse parallel potential paths, including top-down (structure-encouragement) and bottom-up (structure-noticing) searching; a network of ideas with changing importance (affecting and affected by these processes), and a workspace for maintaining the structures and attributes created by these processes. The image creation project uses the same framework on a project that removes much of the complexity of CopyCat by removing almost all facticity and combining the graphical interface to these processes with the problem-solving itself.

5.3 Some Codelets

- Propose a random dot (color and location)
- Look for a line between two points
- Use 2 lines to complete an angle
- Propose a line to complete a polygon
- Change an angle to provide greater symmetry

5.4 Assignments

- Complete codelet-evaluation system and slipnet (2 hr).
- Complete 4 codelets and create 4 more (searching, strength-testing, structure-proposing, and attribute-adding) (4 hr).

5.5 Readings

- "Artificial Intelligence: An Aperçu" by Pamela McCorduck (*Dædalus*, Winter 1988 (*The AARON Project; 15 p.*)
- Excerpt from "The CopyCat Project: A Model of Mental Fluidity and Analogy-Making" by Douglas Hofstadter and Melanie Mitchell, from *Fluid Concepts and Creative Analogies* (*Copycat framework; 20 p.*)
- "Minds, Brains, and Programs" by John Searle (*Chinese room analogy; 10 p.*)