Human-Level AI: Challenges for Machine Learning

Tom Dietterich

School of EECS

Oregon State University

Corvallis, OR 97331

http://www.eecs.oregonstate.edu/~tgd

Three Challenges and Three Challenge Problems

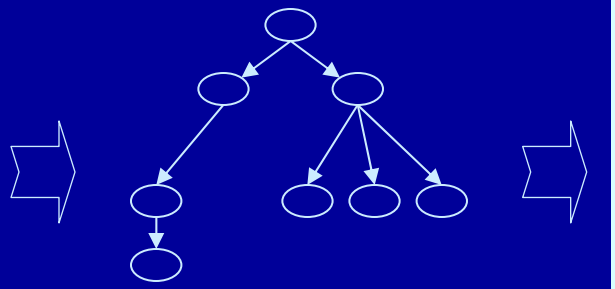
1. Rich “training experiences”
   - See-it/Teach-it challenge
2. Whole agent learning
   - CALO desktop prediction challenge
3. Autonomous formulation and solution of learning problems
   - CALO autonomous learning challenge

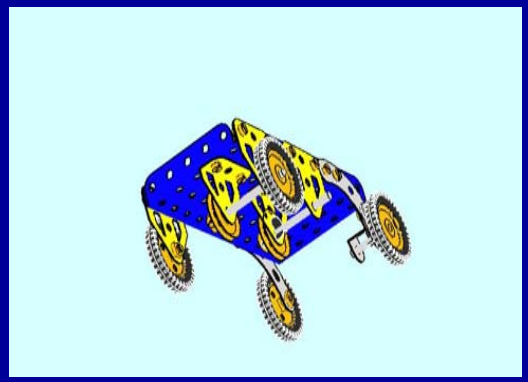
1: Rich Training Experiences

- Most training experiences that people have are much richer than the presentation of labeled feature vectors
  - Example: Tutorial dialogue in which one person shows another person how to assemble a mechanical device
    - The experience is a sequence of actions
    - Natural language commentary highlights key steps, important constraints, reasons behind certain steps, warnings of potential pitfalls, pointers to arbitrary decisions
    - Learner can ask questions
    - Learner can try steps and receive feedback

See-it/Teach-it Challenge

- Domain: Learning to assemble Meccano devices





Human Demonstrates

“see it”

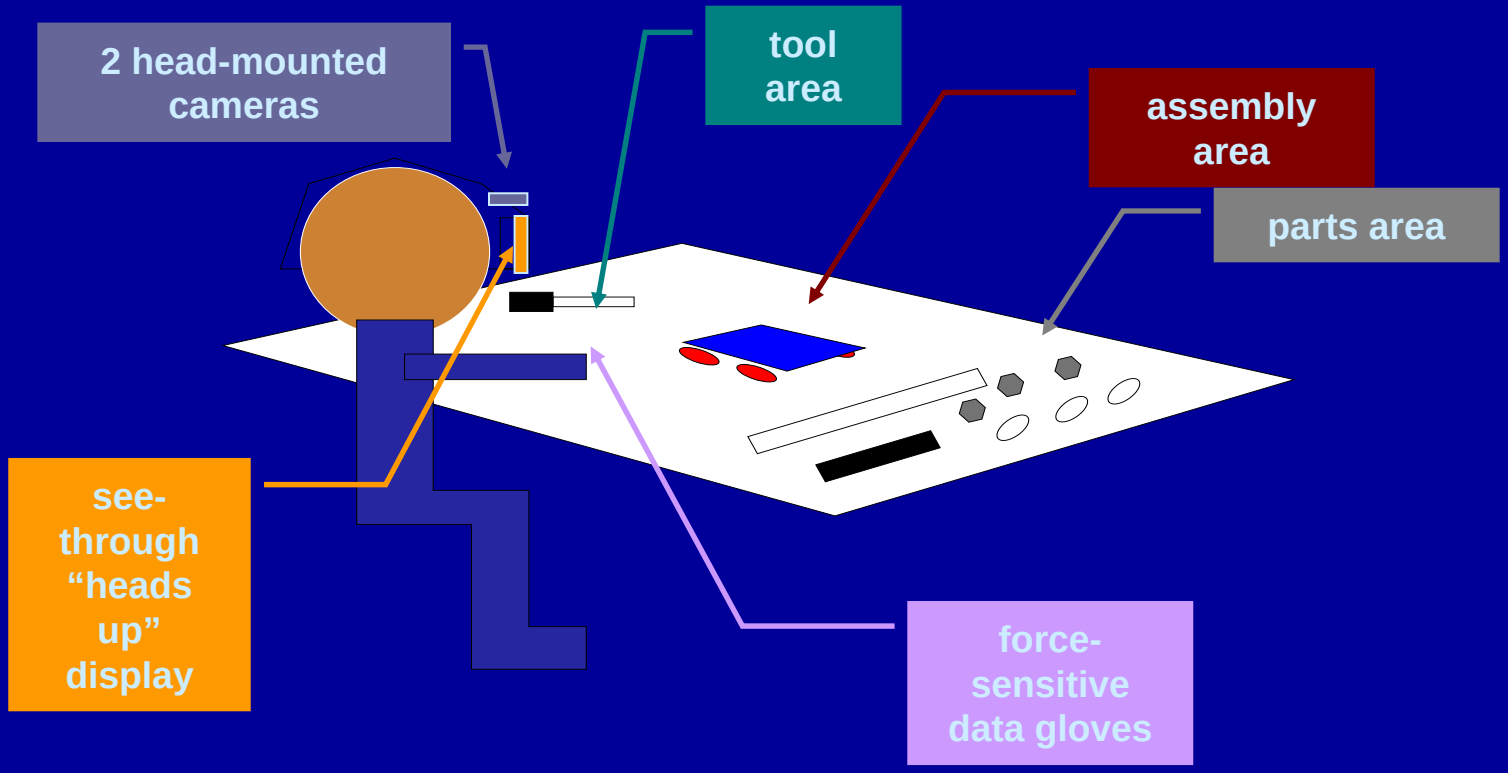
Learner Acquires Generalized Plan

System Instructs Novice

“teach it”

See-it/Teach-it: computer vision + augmented reality

- Used both for learning from expert and for teaching novice
- Technical Challenges:
  - Computer vision + augmented reality display
  - Object recognition and tracking
  - Gesture recognition
  - Track and highlight objects in the real world
  - Provide visualizations of assembly steps



2: Whole Agent Learning

- People's lives are much more complex and elaborate than existing AI agents
- Complex structured states
- Many different people interacting
- Very complex actions
  - physical: locomotion, gestures, manipulation
  - digital: email, IM, PPT, documents, web
  - linguistic: speech, written text
- Very complex goals and utilities

Whole Agent Modeling Challenge

- Given:
  - all electronic activity of a person
    - all computer desktop, email, phone, IM activity
  - video/GPS tracking of the person
  - audio record of person's speech and speech of other agents
- Learn a model sufficient to predict the future activity of the person

CALO: Relational model of the user's life

- Model of projects, people, appointments, files, folders, web sites, IM conversations, job roles, meeting transcripts, action items
- Simultaneous learning of multiple relationships
  - Probabilistic relational models?

3: Autonomous Formulation and Solution of Learning Problems

- Standard Pipeline:
  1. Formulate Problem (inputs, outputs, loss function)
  2. Design Features
  3. Collect Training Data
  4. Choose Model Family and Regularizer
  5. Fit Model
  6. Attach Learned Model to Prediction Opportunities
  7. Attach Predictions to Agent Actions

- Only one of these steps is autonomous currently!
- What would an agent need to know to do this autonomously?

An Example: "Don't let user forget attachments"

1. Identifying the problem: Notice that user often sends two emails in rapid succession. Second one says "Sorry, here is the attachment". Reason that user could save time and avoid embarrassment if we could prevent this.
   - Inputs: email message
   - Output: boolean: message refers to an attachment, but there is no attachment
   - Loss function: 0/1 loss with tunable reject option (because learner may not be perfect
2. Design features: An email message is a kind of document. Bag-of-words features are "good" for documents. Therefore, use bag-of-words representation (with stop list and stemming)
3. Collect training data: Disk contains thousands of previous email messages, many have attachments. Disk contains some email pairs with attachment only on second one. These provide positive and negative examples.

An Example: "Don't let user forget attachments"

4. Choose Model Family: SVMs with dot product kernel (based on previous document classifiers). Use holdout to set regularizer
5. Fit model: SMO
6. Attach to Prediction Opportunities: Hook to the SEND button in Outlook "compose" window
7. Attach to Actions: If predicted positive, pop up window to ask user if he/she wants to attach something to this message. If Yes, abort send. if No, send email. Record new training example in either case.

Autonomous Learning Requirements

- Knowledge of overall goals (be efficient, avoid embarrassment, save money, avoid interruption)
- Recognize situations that could be improved w.r.t. goals if an accurate predictive model could be learned (including cost of prediction errors)
- Assess the feasibility of learning such a model
  - availability of data (including sampling biases)
  - quality of data (predictive power, noise levels)
- Machine learning knowledge
  - definitions of useful features
  - model families
  - regularization strategies
  - learning algorithms (and loss functions)
- Knowledge of the agent's anatomy and physiology
  - how to hook prediction opportunities
  - how to take action on predictions (including good user interfaces)

CALO Autonomous Learning Challenge

????	CALO autonomously does all 7 steps
2007-8	User can specify overall goal, CALO autonomously does the rest
2006-7	User can do all 7 steps via natural user interface
2005-6	Programmer can do all 7 steps using APIs

Summary

- Relational learning
- Learning in the context of rich background knowledge
- Automating much more of the learning process
  - Formulating and solving learning problems based on background knowledge
  - Connecting learned knowledge to action