Combining Inductive and Analytical Learning

- Why combine inductive and analytical learning?
- KBANN: prior knowledge to initialize the hypothesis
- TangentProp, EBNN: prior knowledge alters search objective
- FOCL: prior knowledge alters search operators

Inductive and Analytical Learning

**Inductive learning** | **Analytical learning**
--- | ---
Hypothesis fits data | Hypothesis fits domain theory
Statistical inference | Deductive inference
Requires little prior knowledge | Learns from scarce data
Syntactic inductive bias | Bias is domain theory

What We Would Like

- General purpose learning method:
  - No domain theory → learn as well as inductive methods
  - Perfect domain theory → learn as well as PROLOG-EBG
- Accommodate arbitrary and unknown errors in domain theory
- Accommodate arbitrary and unknown errors in training data

Training Examples

<table>
<thead>
<tr>
<th>BottomIsFlat</th>
<th>Cups</th>
<th>Non-Cups</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>ConcavityPointsUp</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Expensive</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Fragile</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>HandleOnTop</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>HandleOnSide</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>HasConcavity</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>HasHandle</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Light</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>MadeOfCeramic</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>MadeOfPaper</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>MadeOfStyroForm</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

Domain Theory

Cup ← Stable, Liftable, OpenVessel
Stable ← BottomIsFlat
Liftable ← Graspable, Light
Graspable ← HasHandle
OpenVessel ← HasConcavity, ConcavityPointsUp

KBANN

Knowledge Based Artificial Neural Networks

KBANN (data \(D\), domain theory \(B\))
1. Create a feedforward network \(h\) equivalent to \(B\)
2. Use BACKPROP to tune \(h\) to fit \(D\)
Creating Network Equivalent to Domain Theory

Create one unit per horn clause rule (an AND unit)
- Connect unit inputs to corresponding clause antecedents
- For each non-negated antecedent, corresponding input weight $w \leftarrow W$, where $W$ is some constant
- For each negated antecedent, weight $w \leftarrow -W$
- Threshold weight $w_0 \leftarrow -(n - .5) W$, where $n$ is number of non-negated antecedents

Finally, add additional connections with near-zero weights

\[ \text{Liftable} \leftarrow \text{Graspable}, \sim \text{Heavy} \]

Result of Refining the Network

Expensive, BottomIsFlat, MadeOfCeramic, MadeOfStyrofoam, MadeOfPaper, HasHandle, HandleOnTop, HandleOnSide, Light, HasConcavity, ConcavityPointsUp, Fragile

KBANN Results

Classifying promoter regions in DNA (leave one out testing):
- Backpropagation: error rate 8/106
- KBANN: 4/106

Similar improvements on other classification, control tasks.

Hypothesis Space Search in KBANN

Hypothesis Space

EBNN

Explanation Based Neural Network

Key idea:
- Previously learned approximate domain theory
- Domain theory represented by collection of neural networks
- Learn target function as another neural network
**Explanation in Terms of Domain Theory**

Prior learned networks for useful concepts combined into a single target network

- Expensive
- MadeOfCeramic
- MadeOfStyrofoam
- MadeOfPaper
- HasHandle
- HandleOnTop
- HandleOnSide
- Light
- HasConcavity
- ConcavityPointsUp
- Fragile
- Stable
- Graspable
- Liftable
- OpenVessel
- Cup

**TangetProp**

Assume $x$, $f(x)$ and $\frac{\partial f(x)}{\partial x}$ provided as input

Modified objective for gradient descent:

$$E = \frac{1}{2} \sum_i (f(x_i) - \hat{f}(x_i))^2 + \mu \sum_i \left( \frac{\partial f(x_i)}{\partial x_i} - \frac{\partial \hat{f}(x_i)}{\partial x_i} \right)^2$$

where

$$\mu = 1 - \frac{\mu(x_i) - f(x_i)}{e}$$

- $f(x)$ is target function
- $\hat{f}(x)$ is neural net approximation to $f(x)$
- $\mu(x)$ is domain theory approximation to $f(x)$

**FOCL**

- Adaptation of FOIL that uses domain theory
- When adding a literal to a rule, not only consider adding single terms, but also think about adding terms from domain theory
- May also prune specializations generated

**Hypothesis Space Search in TangetProp**

- Hypotheses that maximize fit to data
- Hypotheses that maximize fit to data and prior knowledge

**Search in FOCL**

- Cup ← HasHandle
- Cup ← ~HasHandle
- Cup ← BottomsFlat, Light, HasConcavity, ConcavityPointsUp
- Cup ← BottomsFlat, Light, HasConcavity, ConcavityPointsUp, HandleOnTop
- Cup ← BottomsFlat, Light, HasConcavity, ConcavityPointsUp, ~HandleOnTop
- Cup ← BottomsFlat, Light, HasConcavity, ConcavityPointsUp, HandleOnSide
FOCL Results

Recognizing legal chess endgame positions:
- 30 positive, 30 negative examples
- FOIL: 86%
- FOCL: 94% (using domain theory with 76% accuracy)

NYNEX telephone network diagnosis
- 500 training examples
- FOIL: 90%
- FOCL: 98% (using domain theory with 95% accuracy)