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

<table>
<thead>
<tr>
<th>Inductive learning</th>
<th>Analytical learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypothesis fits data</td>
<td>Hypothesis fits domain theory</td>
</tr>
<tr>
<td>Statistical inference</td>
<td>Deductive inference</td>
</tr>
<tr>
<td>Requires little prior</td>
<td>Learns from scarce data</td>
</tr>
<tr>
<td>knowledge</td>
<td></td>
</tr>
<tr>
<td>Syntactic inductive bias</td>
<td></td>
</tr>
</tbody>
</table>

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

Domain Theory

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

Training Examples

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

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 \( w_0 \leftarrow -(n - .5) \cdot W \), where \( n \) is number of non-negated antecedents

Finally, add additional connections with near-zero weights

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

Result of Refining the Network

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

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

TangetProp

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

Modified objective for gradient descent:

$$E = \sum (f(x) - \hat{f}(x))^2 + \mu \sum \frac{\partial f(x)}{\partial x} \left( \frac{\partial \hat{f}(x)}{\partial x} \right)^2$$

where

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

Hypothesis Space Search in TangetProp

Hypothesis Space

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

Search in FOCL
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)