

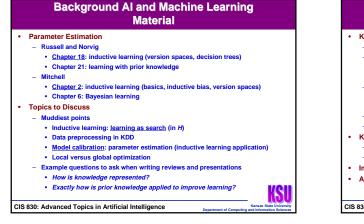
Chown and Dietterich

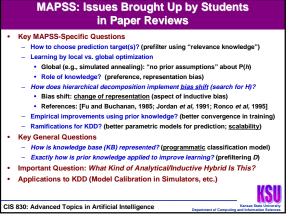
(SI

CIS 830: Advanced Topics in Artificial Intelligence

Lecture Outline Pape Paper: "A Divide-and-Conquer Approach to Learning from Prior Knowledge" Authors: E. Chown and T. G. Dietterich Overview Using prior knowledge as an aid to learning Model calibration problem · Role of prior knowledge in analytical and inductive learning Hierarchical learning system: MAPSS Analytical learning to decompose prediction learning problem sequentially Idea: choose hypothesis language (parameters), examples for subp **Topics to Discuss** How to choose prediction target(s)? Local versus global optimization: how can knowledge make difference? How does hierarchical decomposition implement bias shift (search for H)? Empirical improvements using prior knowledge? Ramifications for KDD? Next Paper: Towell, Shavlik, and Noordewier, 1990 (KBANN) Б

CIS 830: Advanced Topics in Artificial Intelligence





Key Strengths of MAPSS Learning Technique

Strengths

- Prior knowledge led to training convergence
- Previously, could only calibrate 12 of 20 parameters of model (Section 2.2)
 Prior knowledge made it possible to calibrate rest (Section 3.3)
- Idea: analysis of code to produce prior knowledge

 Knowledge-based software engineering (KBSE) concept
- Implement <u>classification model</u> as program
- Use <u>partial evaluation</u> of program to find $x \in D$ for which *few* θ_i are unknown
- Idea: bootstrapped (interleaved inductive, analytical) learning
- Training: "short runs" of global optimization, interleaved with prefiltering of *D* Produces <u>filter models</u> and one example per model (batch of 40)
- Idea: decomposing problems into locally relevant sets of parameters
 <u>Scalability</u> (through divide-and-conquer): relative to θ_l (65 attributes)
- Partitioning problem by <u>partitioning attributes</u> [Hsu, Ray, and Wilkins, 2000]
 Applications to KDD
- Can express many KBs as programs: simulators, classification systems
 - Methods for estimating (e.g., EM) missing values in data
 - Breaking problem into more tractable pieces (more in Paper 8!)

CIS 830: Advanced Topics in Artificial Intelligence

Weaknesses - Still took 3+ months (even using prior knowledge)! • 750K evaluations took 6 CPU weeks (SPARC 2) • 1.5M evaluations in final version - Generality not well established • Under what conditions can we express prediction rules in the imperative programming language used?

Key Weaknesses of MAPSS Learning Technique

- Ramifications for general-case learning applications (e.g., KDD?)
- Typos in section 3.2?
- Unclear Points
- What form of partial evaluation is appropriate for prediction task?
- How to choose the right architecture of committee machine? (e.g., filter models)
- Can technique scale up calibration of broad class of scientific models?
- How to use prior relevance knowledge in KDD?
- Acquisition (<u>a</u>utomatic <u>relevance d</u>etermination, aka ARD) "20 important θ_i"
 Automatic application (stay tuned...)
- How to apply other forms of prior knowledge (constraints, etc.)? Paper 4
- CIS 830: Advanced Topics in Artificial Intelligence Kansas State University
 Department of Computing and Information Sciences

Data Gathering Algorithm

Committee Machine

See Chapter 7, Haykin

- Chapter 7, Mitchell
- Lectures 21-22, CIS798 (http://ringil.cis.ksu.edu/Courses/Fall-1999/CIS798) Idea
- Use experts to preprocess (filter) D or combine predictions
- In this case, 40 experts prefilter D to get n = 40 examples; need 32-36 to agree Intuitive Idea
- Want to use prior knowledge (in form of imperative program) to speed up learning Analyze program: perform partial evaluation using current calibration
- · Prefilter data: find "good operating regions" (classification paths with "few enough" unknown parameters) - Algorithm: technical details
- Need to reduce sensitivity (instability): 1 example per model (of 40) Accumulate 40 "good" training examples

181

CIS 830: Advanced Topics in Artificial Intelligence

Scaling Up KDD **Using Prior Knowledge** MAPSS Problem n = |D| = 40: considered "small" for this problem Not clear how many candidates, but only 5 filter passes suffices <u>Nota Bene: Takes many experts (32-36 out of 40) to get good "consensus"!</u> m = 65 attributes: considered "medium" for this problem (given n) 5 prediction targets 3 leaf area index (LAI) predictions, 1 runoff prediction (numerical) • 1 biome classification (74 possible values) Prior Knowledge: Lessons Learned Previous approaches KBANN: backpropagation in feedforward ANNs using "compiled" constraints FOCL: variant of FOIL (decision trees using first-order logic predicates) Others: qualitative simulation, inductive logic programming (ILP), etc. Problem: lack of scalability

- · Computational limitations of inference (semidecidability of resolution)
- · Intractability of even very restricted learning approaches 15

CIS 830: Advanced Topics in Artificial Intelligence

Course Project: Overview
3 Components - Project proposal (20%, 50 points)
 Implementation (50%, 125 points) Final report (30%, 75 points)
Project Proposal (Due 02/14/2000)
 1-3 page description of project topic, plan Guidelines: next (and suggested topics, tools on course web page)
Implementation Students choice of programming language
- Guidelines: Friday (and on course web page) Final Report
- 4-6 page report on implementation, experimental results, interpretation
- Peer-reviewed (does not determine grade) Reviews graded (short report worth 60 points, reviews worth 15 points)
CIS 830: Advanced Topics in Artificial Intelligence Kansas State University Department of Computing and Information Science

	Course Project: Proposal Guidelines
•	Report Contents (1-3 Pages)
	– Scope: What kind of data will you use?
	– Problem: What problem are you addressing?
	– Methodology: How are you addressing the problem?
•	Scope
	 What data sets will you use?
	– What characteristics of the data are you trying to deal with / exploit?
•	Problem
	 Objective: What KDD problem are <u>you</u> trying to solve?
	 Performance element: What is the problem-solving component of your KDD system?
	– Evaluation: How will you measure success?
•	Methodology
	- Implementation: What will you implement? (general statement, not specification
	- Tools: What programming languages and KDD tools will you use?
cis	830: Advanced Topics in Artificial Intelligence Kansas State University Computing and Information Scie

Terminology		
Inductive Learning		
- Prior knowledge		
Declarative: expressed in assertions (e.g., FOPC)		
Procedural: expressed in imperative statements		
Eunctional: expressed as functions (e.g., higher-order) and relations		
<u>Taxonomic</u> : expressed as classification hierarchy		
Inductive biog		

- ctive bias
- <u>Representation bias</u>: expressed by *H*, hypothesis space (language)
- Preference bias: expressed by L, learning algorithm
- Change of representation: transformation from H into H (form of bias shift)
- Bias shift: change in inductive bias (representation or preferences)
- Divide-and-Conquer Approaches to Learning
- Hierarchical learning systems: decompose problem according to attributes, examples, etc.
- Committee machines: combine outputs of multiple expert "modules" 181

CIS 830: Advanced Topics in Artificial Intelligence

 Key Points Covered - Using prior (declarative) knowledge as an aid to learning Hierarchical learning system: MAPSS Bias shift through systematic problem decomposition Idea: choose hypothesis language (parameters), examples for subproblems Discussion Topics - Local versus global optimization: knowledge as bias (control of search over H)

Summary Points

- Scalable KDD: hierarchical decomposition using relevance knowledge
- · Prior knowledge in form of classification program
- · Developing relevance knowledge using partial evaluation - Choosing prediction targets in KDD: general filtering problem
- Next Paper

- Towell, Shavlik, and Noordewier, 1990

- "Knowledge-Based Artificial Neural Networks (KBANN)": constraints in feedforward ANN learning
- CIS 830: Advanced Topics in Artificial Intelligence

18