Educational Technologies
WS2008

Student Modeling
and Competencies

Source: Erica Melis
Approximate Plan of the Course

- 28.10.2008 Introduction and overview
- 04.11.2008: Intelligent tutoring systems (1) - Cognitive Tutors
- 11.11.2008: Intelligent tutoring systems (2) – ActiveMath
- 18.11.2008: Student modelling
- 25.11.2008: Student modelling
  : Pedagogical components, instructional planning
- 02.12.2008: Error diagnosis and feedback
- 09.12.2008: Meta-cognitive support (1) – Help
- 16.12.2008: Meta-cognitive support (2)
- 13.01.2009: Collaborative learning technologies
- 20.01.2009: Multi-Media Learning principles
- 27.01.2009: Web-based systems
- 03.02.2009: Educational data mining
- 10.02.2009: Project presentations by students
Overview, Student Modelling

- **General**
- **What is modelled**
  - input and output.
- **Techniques**
  - Bayesian networks
  - Bayesian updating
  - Dynamic Bayesian Nets and Decision Networks
  - Affective Computing
  - Partial Order Knowledge Spaces
  - Stereotypes
  - Social voting
  - Hidden Markov Models

- **Student Modeling in ActiveMath**
Reminder: Generic ITS Architecture

- Student KR
- Problem Solver
- Problem Selector
- Curriculum Planner
- Intelligent Tutorial Component
- Solution Graph
- Curriculum
- Solution Evaluator
- Student Model
- Feedback Generator
- Interaction History
- Action Interpreter
- Graphical user interface
- Domain KR
- Intelligent Tutorial Component
- Student
- Problem
- Solver
- Selector
- Planner
- Graph
- Curriculum
- Evaluator
- Model
- Feedback
- Interaction
- History
- Action
- Interpreter
- Graphical
- user
- interface
- Domain
- KR
Static Student Modelling

- Language
- Country / region / community of practice
- Educational context
- Technical context
- Social context
Dynamic Student Modeling

- **Competency assessment**
- **Plan recognition**
- **Affect diagnosis/prediction**
  - Confused
  - Interested
  - Bored
  - Frustrated
  - Anxious
  - Eureka
  - Persisting
  - Neutral
- **Motivation and attitude monitoring**
  - Learning goal, self-efficiency...
Systematic Approach

input

Student Model

relevant output
Purpose Adaptation >> SM Output >> Input

- **To personal variables:**
  - **To dynamic variables:** content selection and sequencing, help, feedback, suggestions, autonomy, group formation, pedagogical agent’s behaviour
  - **To static variables:** Language, curriculum, rendering of formulae, politeness degree

- **To technical environment**
  - rendering(browser, size of screen), selection of LOs (bandwidth, size of screen) (rendering, selection), modi(audio?)

- **To social context**
  - Type of collaboration (distance l. vs classroom), scenario (remedy course), tools
Inputs for Student Model

- **Behavioral data – content metadata**
  - Performance
  - number of tasks finished
  - response to test items (hesitation)
  - time-on-task
  - # help or other requests …

- **Sensor data:**
  - Facial expression, skin conductance, heart frequency, posture

- **Self-reports:**
  - questionnaire
  - on-task-report 😊 😞
  - open Learner Model

- **Teacher input**

- **Context data**
Definition von Änderung

Gegeben sei eine variable Größe $x$ und ein Ausgangswert $x_0$ dieser Größe. Unter der Veränderung oder kurz Änderung von $x$ versteht man die Differenz $x-x_0$. Differenzen werden gewöhnlich mit dem griechischen Buchstaben $\Delta$ bezeichnet, wir haben daher:

$$\Delta x = x - x_0.$$  

Funktionswertänderungen

Der Wert der Funktion $f(x) = x^2$ ändert sich zwischen $x_0 = 1$ und $x = 4$ um den Wert

$$\Delta f = f(x) - f(x_0) = f(4) - f(1) = 4^2 - 1^2 = 15.$$  

Berechnen von Änderungen

Eine kleine Rechenaufgabe.

Aufgabe starten
Output: Individual Variables

- **Cognitive (actual)**
  - Competencies
  - Cognitive/learning style*
  - Problem solving strategies, preferences
  - Meta-cognitive skills

- **Psychological (mental)**
  - Interests
  - Preferences, learning style*

- **Situational**
  - Exploratory behaviour
  - Goals: performance vs learning
  - Autonomy and approval needs

- **Affective, motivation**

- **Personal traits**
  - Blind, working memory capacity, attention span, reading capability
## Output: Competencies

<table>
<thead>
<tr>
<th>PISA [2003]</th>
<th>Bloom's taxonomy [1956]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solve</td>
<td>Knowledge</td>
</tr>
<tr>
<td>Argue</td>
<td>Remember, recognize</td>
</tr>
<tr>
<td>Think mathematically</td>
<td>Comprehension</td>
</tr>
<tr>
<td></td>
<td>▶ Formulate</td>
</tr>
<tr>
<td></td>
<td>▶ Interpret, translate, describe in own words</td>
</tr>
<tr>
<td></td>
<td>▶ Generalize</td>
</tr>
<tr>
<td>Model</td>
<td>Application</td>
</tr>
<tr>
<td></td>
<td>▶ Encode</td>
</tr>
<tr>
<td></td>
<td>▶ Problem solving, use facts</td>
</tr>
<tr>
<td></td>
<td>▶ Decode</td>
</tr>
<tr>
<td>Represent</td>
<td>Analysis</td>
</tr>
<tr>
<td>Communicate</td>
<td>▶ Deconstruct, structure, find motivation …</td>
</tr>
<tr>
<td>Tools</td>
<td>Synthesis</td>
</tr>
</tbody>
</table>
Towards Standard Competency System

[MelisFaulhaber..2008]

- **Interoperable / comparable**
  - Exchange competency data btw. systems
  - Reuse of learning objects

- **Across domains**
  - Separate domain dependent/independent parts

- **Support all intended usages**
  - Assessment, curriculum matching, …

- **Capture meta-cognitive competencies**
Contd: 2-dimensional Competencies

[AndersonKrathwohl2001]

- **Elementary competency**
  
  $c = (p, k)$

  - **Domain dependent**
  - **Domain independent**

- $p$: cognitive process (e.g. apply algorithm)
- $k$: knowledge element (e.g. def of fraction, sum rule, add fractions)
### Process Hierarchy for Mathematics

<table>
<thead>
<tr>
<th>Remember</th>
<th>Recognize, Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Represent</td>
<td>Interpret, Exemplify, Transform, Summarize</td>
</tr>
<tr>
<td>Compare</td>
<td>Find commonalities, Find differences, Classify, Infer, Order</td>
</tr>
<tr>
<td>Solve</td>
<td>Estimate, Apply algorithm, Apply tool</td>
</tr>
<tr>
<td>Analyze</td>
<td>Check, Differentiate, Organize, Attribute</td>
</tr>
<tr>
<td>Model</td>
<td>Decode, Encode, Generate, Produce</td>
</tr>
<tr>
<td>Communicate</td>
<td>Describe, Explain, Critique, Prove</td>
</tr>
<tr>
<td>Meta-cognition</td>
<td>Reflect, Help seeking, Search for information, Detect errors, Plan, Self-monitor, Self-explain</td>
</tr>
</tbody>
</table>

**Hierarchy of**

- 8 main processes
- 33 subprocesses

**Extensible**

- Add another subprocess
## Competency Hierarchy for Mathematics

<table>
<thead>
<tr>
<th>Competency</th>
<th>Sub-Competencies</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remember</td>
<td>Recognize, Recall</td>
<td></td>
</tr>
<tr>
<td>Represent</td>
<td>Interpret, Exemplify, Transform, Summarize</td>
<td></td>
</tr>
<tr>
<td>Compare</td>
<td>Find commonalities, Find differences, Classify, Infer, Order</td>
<td></td>
</tr>
<tr>
<td>Solve</td>
<td>Estimate, Apply algorithm, Apply tool</td>
<td>Find the error in $1/3 + 1/2 = 2/5$</td>
</tr>
<tr>
<td>Analyze</td>
<td>Check, Differentiate, Organize, Attribute</td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td>Decode, Encode, Generate, Produce</td>
<td></td>
</tr>
<tr>
<td>Communicate</td>
<td>Describe, Explain, Critique, Prove</td>
<td></td>
</tr>
<tr>
<td>Meta-cognition</td>
<td>Reflect, Help seeking, Search for information, Detect errors, Plan, Self-monitor, Self-explain</td>
<td></td>
</tr>
</tbody>
</table>

Source: Erica Melis

Educational Technologies WS 2008/09
Assignment of Competencies to Exc

**Metadata:**
Pairs of (process, knowledge)

- Clear Separation
  - Domain ontology
  - Processes taxonomy
  - Content

⇒ Interoperability through:
  ontology / taxonomy alignment
Student Modelling Techniques

- Bayesian networks
- Bayesian updating
- Dynamic Bayesian Networks
- Affective computing, machine learning
- Data-driven (KS, IRT)
- Stereotypes (deductive)
- Social voting
Bayes Networks, overview

- Conditional probabilities, Conditional independence
- Structure of Bayesian networks
- Inference in Bayesian networks
- Applications
Conditional probability (1)

Conditional (or posterior) probabilities

Example: \( P(\text{cavity} \mid \text{toothache}) = 0.8 \)
i.e., given that \( \text{toothache} \) is all I know

\textbf{NOT} “if \( \text{toothache} \) then 80% chance of \( \text{cavity} \)”

If we know more, e.g., \( \text{cavity} \) is also given, then we have

\[ P(\text{cavity} \mid \text{toothache}, \text{cavity}) = 1 \]
Conditional probability (2)

**Definition** of conditional probability:

\[ P(a \mid b) = \frac{P(a \land b)}{P(b)} \text{ if } P(b) \neq 0 \]

Product rule gives an alternative formulation:

\[ P(a \land b) = P(a \mid b)P(b) \]

A general version holds for whole distributions, e.g.

\[ P(\text{Weather}, \text{Cavity}) = P(\text{Weather} \mid \text{Cavity}) P(\text{Cavity}) \]
Conditional probability (3)

Chain rule is derived by successive application of product rule:

\[
P(X_1, \ldots, X_n) = P(X_1, \ldots, X_{n-1})P(X_n \mid X_1, \ldots, X_{n-1}) \\
= P(X_1, \ldots, X_{n-2})P(X_{n_1} \mid X_1, \ldots, X_{n-2})P(X_n \mid X_1, \ldots, X_{n-1}) \\
= \ldots \\
= \prod_{i=1}^{n} P(X_i \mid X_1, \ldots, X_{i-1})
\]
Inference by enumeration: event probability

Start with the joint distribution:

<table>
<thead>
<tr>
<th></th>
<th>toothache</th>
<th>¬ toothache</th>
</tr>
</thead>
<tbody>
<tr>
<td>catch</td>
<td>.108</td>
<td>.072</td>
</tr>
<tr>
<td>¬ catch</td>
<td>.012</td>
<td>.008</td>
</tr>
<tr>
<td>cavity</td>
<td>.016</td>
<td>.144</td>
</tr>
<tr>
<td>¬ cavity</td>
<td>.064</td>
<td>.576</td>
</tr>
</tbody>
</table>

For any proposition $\Theta$, sum the atomic events where it is true:

$$P(\Theta) = \sum_{\{\omega: \omega = \Theta\}} P(\omega)$$

$P(toothache) = 0.108 + 0.012 + 0.016 + 0.064 = 0.2$
Inference by enumeration: cond probability

Start with the joint distribution:

<table>
<thead>
<tr>
<th></th>
<th>toothache</th>
<th>¬ toothache</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>catch</td>
<td>¬ catch</td>
</tr>
<tr>
<td>cavity</td>
<td>.108</td>
<td>.012</td>
</tr>
<tr>
<td>¬ cavity</td>
<td>.016</td>
<td>.064</td>
</tr>
</tbody>
</table>

Conditional probabilities, e.g.:

\[
P(\neg cavity | \text{toothache}) = \frac{P(\neg cavity \land \text{toothache})}{P(\text{toothache})} = \frac{0.016 + 0.064}{0.108 + 0.012 + 0.016 + 0.064} = 0.4
\]
Inference by enumeration, contd.

Typically, we are interested in

the posterior joint distribution of a query variables $Y$
given specific values $e$ for the evidence variables $E$

Let the hidden variables be $H = X - Y - E$

Then the required summation of joint entries is done by summing out the hidden variables:

$$P(Y|E = e) = \alpha P(Y|E = e) = \alpha \sum_h P(Y,E = e, H = h)$$

Exclude irrelevant variables and recursively compute
Bayesian Networks

- **Nodes**: set of random variables $X_1, X_2, \ldots, X_n$
- **Links**: probabilistic dependencies among variables

- **CPTs**: quantify the (conditional) dependencies

```
A  B
\[ P(A) = 0.01 \]
\[ P(B) = 0.02 \]

C
\[ P(C | A, B) = 0.95 \]
\[ P(C | A, \neg B) = 0.94 \]
\[ P(C | \neg A, B) = 0.29 \]
\[ P(C | \neg A, \neg B) = 0.001 \]

D  E
\[ P(D | C) = 0.90 \]
\[ P(D | \neg C) = 0.05 \]
\[ P(E | C) = 0.70 \]
\[ P(E | \neg C) = 0.01 \]
Bayesian Networks, Example

I’m at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn’t call. Sometimes it’s set off by minor earthquakes. Is there a burglar?

Variables: Burglar, Earthquake, Alarm, JohnCalls, MaryCalls

Network topology reflects “causal” knowledge:

- A burglar can set the alarm off
- An earthquake can set the alarm off
- The alarm can cause Mary to call
- The alarm can cause John to call
Example contd.

J and M are **conditionally independent** given A…
Irrelevant variables

Consider the query \( P(JohnCalls \mid \text{Burglary} = \text{true}) \)

M is irrelevant to the query:

Thm1: Y is irrelevant unless \( Y \in \text{Ancestors}(\{X\} \cup E) \)

Here, \( X = \text{JohnCalls} \), \( E = \{\text{Burglary}\} \), and
\( \text{Ancestors}(\{X\} \cup E) = \{\text{Alarm, Earthquake}\} \)

so M is irrelevant
Bayesian Networks in Lumiere [Horwitz et al. 1998]

User expertise

Task difficulty

User needs assistance

User distracted

Pause after activity

Recent menu surfing

http://research.microsoft.com/~horvitz/lum.htm
BN for predicting student performance

Relevant for ideal solution $c,p$

Relevant for student solution $c,p$

Mastered($c$)

Performance $c,p$

Used for selecting problem difficulty in SQL Tutor

<table>
<thead>
<tr>
<th>YY</th>
<th>YN</th>
<th>NY</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>good</td>
<td>1-slip</td>
<td>guess</td>
<td>0</td>
</tr>
<tr>
<td>bad</td>
<td>slip</td>
<td>1-guess</td>
<td>0</td>
</tr>
<tr>
<td>not relev</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Source: Erica Melis Educational Technologies WS 2008/09
Andes1 Network Structure (Static Part)  
[vanLehn et al1997]

▶ Rule nodes

▶ P(R = T): probability that the student knows the rule

▶ Context rule nodes

▶ P(CR = T): probability that the student can use the rule in the corresponding context/exercise

\[
P(CR_i = T \mid R = T) = 1
\]

\[
P(CR_i = T \mid R = F) \text{ estimates the level of difficulty of context } i
\]
Bayes Net Application for Situational Variables

[Situational variables based on information from system]

- student interest; student confidence; student effort; student aptitude; level of student hesitation; level of student achievement

[Numeral Autonomy and Approval values based on the values of the situational variables:]

- level of recommended guidance to be given to the student (Autonomy)
- level of explicit approval recommended for a student (Approval)
Bayes Net for Motivational Variables, simpl.

- **autonomy**
- **approval**
- **confidence**
- **hesitation**
- **initiative**
- **granularityOfSteps**
- **achievement**
- **correctness**

**Factors**:
- **interest**
- **effort**
- **ability**
- **confidence**
- **hesitation**
- **initiative**
- **granularityOfSteps**
- **achievement**
- **correctness**

**Source**: Erica Melis Educationa l Technologies WS 2008/09
Creation of Bayes Nets from Data

Correlations: observable – non-observables

Learning factor

TimeOnHint

RatioHelpedProblems

HintsSeenPerProblem

UsedHeadphone

ProblemsPerMinute

TimeBetweenAttempt

confirmatoryHelpAttitude

seriousTryLearnAttitude

getItOverAttitude

fearWrongAnswer

challenge

Like

Gender

Helpful

Source: Erica Melis

Educational Technologies WS 2008/09
Creating BN from Data: CPTs: Example

<table>
<thead>
<tr>
<th>fearWrong</th>
<th>challenge</th>
<th>timeBetweenAttempts</th>
<th>cases</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>False</td>
<td>Low</td>
<td>43</td>
<td>0.64</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>High</td>
<td>24</td>
<td>0.36</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>Low</td>
<td>35</td>
<td>0.42</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>High</td>
<td>48</td>
<td>0.58</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>Low</td>
<td>8</td>
<td>0.50</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>High</td>
<td>8</td>
<td>0.50</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>Low</td>
<td>7</td>
<td>0.32</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>High</td>
<td>15</td>
<td>0.68</td>
</tr>
</tbody>
</table>
Andes1 Bayes Net: Model Tracing

Problem 1

Goal1

Context-Rule1

Fact1

Fact2

Rule

Problem 2

Fact3

Context-Rule2

Goal2

Fact4
Model Tracing in Andes1: Usage

- Assess intention -> useful hints
- if error + help request -> best explanation
- low mastery of important rules -> mini-lesson
Andes1’ Select a Hint Algorithm

► Step 1: Goal inference
  ▶ Start with last observed student entry
  ▶ Find closest unsatisfied goal node
  ▶ At decision points, choose highest probability node (most likely goal)

► Step 2: Prediction
  ▶ Start from goal found in step 1
  ▶ Find path from goal to “stuck node” (p<0.8)
  ▶ At decision points, choose lowest probability node (most help needed)
Model Tracing: **PACT/cognitive tutors**

- **EXPERT problem solving space** (production rules)
  - includes rules and buggy rules
- match student‘s results with nodes in space
  - hypothesize next possible step
  - diagnoses errors
Problems with Model Tracing

end of lecture1, exclude HMM

- Correct domain rules provide *positive* evidence only
- for *negative* evidence buggy rules needed
- assessors most accurate if
  - student kept along a given solution path
  - Student provides every (intermediate) step
Bayesian Knowledge Tracing

\[ p(K_n) = p(K_{n-1} | \text{evidence}) + (1 - p(K_{n-1} | \text{evidence})) \times p(T) \]

- \( p(K_0) \): initial learning (a priori) probability of rule
- \( p(T) \): transition probability following an opportunity of application.

\[ p(\text{correct}) = p(K) \times (1 - p(S)) + (1 - p(K)) \times p(G) \]

- \( p(G) \): guess
- \( p(S) \): slip
Dynamic Bayesian Networks

\[ P(K_0) \]

\[ BN_0 \rightarrow BN_1 \]

\[ BN_1 \rightarrow BN_n \]

Time \( t_i \)
Dynamic Bayesian Networks [ChangBeckMostowCorbett2008]
(with BNT package)

Student Knowledge $K_0$ -> Student Knowledge $K_1$ -> Student Knowledge $K_n$

Tutor Intervention

Student Performance $C_0$ -> H1

H1 -> Student Performance $C_1$ -> Hn

guess slip

learn, forget

Student Performance $C_0$ -> Tutor Intervention

H1 -> Tutor Intervention

Hn -> Tutor Intervention
 KT as Special Case of DBN

\[
\begin{align*}
\text{learn}: P(K_n | \text{not } K_{n+1}) \\
\text{forget}: P(\text{not } K_n | K_{n+1})
\end{align*}
\]

\textbf{guess}: \ P(C_n | \text{not} K_n) \\
\textbf{slip}: \ P(\text{not} C_n | K_n)

\[
\begin{align*}
\text{Student Knowledge } & K_0 \\
\text{Student Knowledge } & K_1 \\
\text{Student Knowledge } & K_n
\end{align*}
\]

\[
\begin{align*}
\text{Student Performance } & C_0 \\
\text{Student Performance } & C_1 \\
\text{Student Performance } & C_n
\end{align*}
\]
Extended Knowledge Tracing with DBN

[MostowZhang2008]

Student Knowledge ($K_i$) → Practice Mode $P_{Mi}$ → Credited $C_i$ → Binary 0/1

Student Knowledge ($K_{i+1}$) → Practice Mode $P_{Mi+1}$ → Credited $C_{i+1}$

Swap Reading
Affective Computing:
Input: sensor Data, Output: Affective Learning-Related Variables

- Confused
- Interested
- Bored
- Frustrated
- Anxious
- Eureka
- Neutral

- Disagreeing
- Agreeing
- Persisting
Affective Computing: ALC System Architecture

System Architecture with sensors listed from right to left: video camera, Blue Eyes camera, pressure mouse, skin conductance sensor, posture chair.

Source: Erica Melis  Educational Technologies WS 2008/09
Affective Computing,
Input: sensor data, Output: affective variables
Affective Computing

Input: sensor data, Intermediate Output: posture

Source: Erica Melis  Educational Technologies WS 2008/09
Input: sensor data, Intermediate Output: smile, blink, shake
Affective Computing
Input: sensor data, Output: affective var. [Picard et al 2008]
Machine Learning for Pre-Frustration

- **Input:** intermediate data
- **Output:** classification of (binary) pre-frustration
- **Multiple sensors yield robustness**

![Table showing accuracy of different methods]

[KapoorBurlesonPicard2008]
# Affective Computing: AutoTutor Rules

[D'Mello..Picard,Graesser 2008]

<table>
<thead>
<tr>
<th>Student Model</th>
<th>Tutor Action</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Current Emotion</strong></td>
<td>Feedback</td>
</tr>
<tr>
<td>boredom, confusion, frustration</td>
<td>positive, neutral, negative</td>
</tr>
<tr>
<td><strong>Classification Confidence</strong></td>
<td>Empathetic and motivational statement</td>
</tr>
<tr>
<td>high or low</td>
<td></td>
</tr>
<tr>
<td><strong>Previous Emotion</strong></td>
<td>Next Dialogue Move</td>
</tr>
<tr>
<td>boredom, confusion, frustration</td>
<td>hint, pump, prompt, splice, assertion</td>
</tr>
<tr>
<td><strong>Global Student Ability</strong></td>
<td>Facial Expression</td>
</tr>
<tr>
<td>high or low</td>
<td>surprise, delight, compassion, skeptical</td>
</tr>
<tr>
<td><strong>Quality of Current Answer</strong></td>
<td>Speech intonation</td>
</tr>
<tr>
<td>high or low</td>
<td>pitch, intensity, speech rate, etc</td>
</tr>
</tbody>
</table>

Production rules to respond to learners’ affective and cognitive States
Affective Computing: AutoTutor Rules

[D'Mello..Picard,Graesser 2008]

Acoustic-prosodic correlates of tutors emotional expressions

<table>
<thead>
<tr>
<th>Affective State</th>
<th>Pitch Range</th>
<th>Pitch Level</th>
<th>Speech Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surprise/Delight</td>
<td>Wide</td>
<td>Very High</td>
<td>Fast</td>
</tr>
<tr>
<td>Empathy</td>
<td>Narrow</td>
<td>Low</td>
<td>Slow</td>
</tr>
<tr>
<td>Skeptical</td>
<td>Narrow</td>
<td>High</td>
<td>Slow</td>
</tr>
<tr>
<td>Disappointment</td>
<td>Narrow</td>
<td>Low</td>
<td>Slow</td>
</tr>
</tbody>
</table>

[Delight] [Skeptical]

[Compassion] [Disappointment]
Knowledge Spaces: Data-Driven Approach

Knowledge space theory [Falmagne1999]

- Originally for adaptive testing psychometric methods
- Item to item node structure
- Compute individual knowledge space
  - Compute potential boundaries for adaptive testing
Item to Item Node Structure (POKS)

- DAG of exercises (items)
- Edges point to prerequisites
- \( a \rightarrow c \): mastery of \( c \) will precede mastery of \( a \). If student succeeds with \( a \) then with \( c \) too
- Likely partial ordering which items learned
Knowledge Space, POKS inference

- Failure at a node implies: Items that depend on this node are not yet mastered (negative downward propagation)

- Mastery of a node implies: Mastery of prerequisites (positive upward propagation)
Knowledge Space Propagation

- If student solves B, prerequisite D can be assumed to be mastered
- If student fails at C, depending item A is most likely not yet mastered
Knowledge Spaces

- Individual’s knowledge state = subset of items (mastered)
- Possible knowledge states: \( \{\emptyset, \{d\}, \{c,d\}, \{b,d\}, \{b,c,d\}, \{a,b,c,d\}\} \)
- Determine mastery of one ability dimensions
- Knowledge state determines which state to move to
  other knowledge states unlikely
Knowledge Space Inference

- **Learning item-to-item structure:**
  - POKS: structural induction, pairwise analysis:
    1. $P(B|A) \geq p(\text{mastered})$
    2. $P(\neg A|\neg B) \geq p(\text{not mastered})$
    3. $P(B|A) \neq P(B) \text{ (dependence)}$

- With network compute probability of success for items from partial evidence with Bayesian posterior $P(H | E_1,E_2...)$

- **Compute from probability distribution:** most likely next knowledge state
  - interest is in predictive power
Stereotypes

- Stereotype: body + set of triggers
- St. may be arranged in hierarchies
- (multiple) inheritance

Membership in (several) stereotypes

- Trigger
- (possible) inheritance

Actions and self-ass

Decision-Relevant facts

Source: Erica Melis  Educational Technologies WS 2008/09
Stereotypes: Rules und Usage

▶ **Trigger rules** *(Cond Trigger)*
  ▶ if self-assessed 'expert' THEN stereotype = Expert
  ▶ if at least 20 topics K=excellent THEN stereotype=Expert

▶ **Rules for evaluation and presentation** *(Cond Scoring)*
  ‘Easy-Topic-for-Beginners’
  ▶ Beginner AND diff(t)=simple THEN add 10 points to appr.score(t)
  ‘Avoid-Known-Topic’
  □ If user knows Topic t THEN substract 20 points from appr.score(t)
Welcome to iHelp!

What you are seeing is a new release of the I-Help system - known by its creators in the Computer Science Department at the U of S as iHelp2 (pronounced "I Help Too"). While maintaining much of the functionality of the original I-Help, this version has been re-built to try to be a simple, intuitive, and powerful discussion medium and at the same time scalable and database independent. Please note that there is still work to be done to test and complete this version, so you as a user have a chance to be part of the software development process.

Welcome!

Welcome to iHelp Chat 2.0!
To join a channel, type /join channelname

Collene
Social Voting, i-Help [Greer, McCalla, Bull, Vassileva 2000]
### Social Voting Techniques, Multi-Agent: I-Help

**Student model information:**

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Sources of information</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>User</td>
<td>Peers</td>
<td>I-Help Private</td>
<td>I-Help Public</td>
</tr>
<tr>
<td>knowledge</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
</tr>
<tr>
<td>interests</td>
<td>•</td>
<td></td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>eagerness</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>helpfulness</td>
<td>•</td>
<td></td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>readiness (online)</td>
<td>•</td>
<td></td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>cognitive style</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
</tr>
<tr>
<td>preferences in helper</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
</tr>
<tr>
<td>preferred people</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>banned people</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>banned topics</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>help-load</td>
<td>•</td>
<td></td>
<td></td>
<td>•</td>
</tr>
</tbody>
</table>
Helper model and ranking of helpers by agent

<table>
<thead>
<tr>
<th></th>
<th>Prof 1</th>
<th>Peer 1</th>
<th>Prof 2</th>
<th>TA 1</th>
<th>TA 2</th>
<th>Peer 2</th>
<th>Peer 3</th>
<th>Peer 4</th>
<th>Peer 5</th>
<th>Peer 6</th>
<th>Peer 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>eager</td>
<td>5.45</td>
<td>5.86</td>
<td>4.8</td>
<td>8.36</td>
<td>4.55</td>
<td>6.58</td>
<td>6.36</td>
<td>6.1</td>
<td>5.49</td>
<td>5.9</td>
<td>6.13</td>
</tr>
<tr>
<td>helpful</td>
<td>9.69</td>
<td>9</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>self eval</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>8</td>
<td>10</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>sys eval</td>
<td>10</td>
<td>4.75</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>ready</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Helpfulness << was helper helpful?
  was helper easy to understand?
Frequency and speed of response

Eagerness << logins
  postings in discussions
  updating profile to agent
ActiveMath adapts to:

- Language, mastery level, competencies, student input, goals, prerequisites, field of interest, school type, field, browser

- **Model variables**
  - Static: Learning context, interests, language…
  - Dynamic: *Competencies*

- **Evidence for competencies:** Exercise input

- **Uses IRT** (Item Response Theory) + **TBM** (Transferable Belief Model) for competency updating

- Update propagation

- Constructs topic map from content
IRT Model relates probability of correct answer to

- person's proficiency (mastery)
- exercise (item) difficulty...

\[ p_i(\theta) = c_i + \frac{(1 - c_i)}{1 + e^{-Da_i(\theta - b_i)}} \]

- \( \theta \) test person proficiency (mastery)
- \( a_i \) item discrimination factor
- \( b_i \) item location (difficulty)
- \( c_i \) item guessing probability
- \( D \) const.
Compute IRT Parameters

\[ \theta \] test person proficiency (mastery)

\[ a_i \] item discrimination factor

\[ b_i \] item location (difficulty)

\[ c_i \] item guessing probability

\[ D \] const.

metadata in content

- Difficulty
IRT contd.

- **Estimate** mastery

  - TBM belief system to combine evidences

**Challenges**

- Realistic estimates
  - 1 very simple exercise solved
    - common IRT -> very high proficiency -> very hard exercise

- Derive meaningful estimations with little evidence

- Permit estimation to account for pedagogical aspects
  - Do not present too simple/difficult exercises
Deriving Belief Masses with TBM

• Hypotheses in TBM
  – 35 atomic hypotheses:
    \[ H_0 \leq i \leq 34 : \text{"Learner has mastery } m = i \cdot 0.03\"
    
    e.g. \( H_4 \) : has mastery 12%
    \[ H_5 : \] 15%
    \[ H_6 \] : 18% …

• Assign weights to hypotheses \( H_i \) from function value of IRT sigmoid function

Source: Erica Melis  Educational Technologies WS 2008/09
Workflow of Estimating Mastery Value

User action

Attribute to competency

Create new competency

(Re-)estimate masteries

Interpret evidences: quantify probabilities (IRT)

Derive belief masses from probabilities

Combine evidences in TBM
Exercise 1

Metadata:
- difficulty: simple_conceptual, easy
- for: D

Courses:
- C
- B
- D
- F
- E
- H
- G
- I
- J

Prerequisites:
- C for B
- D for E
- F for G

Direct evidences:
- D

Indirect evidences:
- C for B
- B for D
- F for G

Updating and propagation
SLM in ActiveMath

ActiveMath Components

Communication Infrastructure

Learner Model

Content and Metadata-definitions

Exercise System

Course Planner

Open LearnerModel

EventSystem

LernerModel Interface

SLM

Misconceptions

Mastery Estimations

Competency System(s)

Content
Test of Student Models

- Never fully reliable...
- Empirical test with users
  - protocols..teacher assessors
- Introspectable and modifiable user models
  - student/teacher revises
  - dialogue about student‘s beliefs
- Test with artificial users
Conclusion: Putting the Model to Use

Copyright © 1997 United Feature Syndicate, Inc.
Redistribution in whole or in part prohibited
≈ Wizard of Oz Experiments by students

- Make up a problem to be solved by student actor
- Student model actor and adaptation actor determine output variables of student model
- Student model actor determines input variables and how to get their values (from student actions, sensing, asking the student, etc)
- Adaptation actor adapts feedback etc to student model
- Together write a protocol and analyze
  - Describe data sources, identify topics, goals, identify student knowledge, identify help and hints
- Report in Wiki