Logic, Computability, and Cognition

Lecture 1: Logic & Cognition. Introduction and Some Examples

Jakub Szymanik   Nina Gierasimczuk

Institute for Logic, Language and Computation
University of Amsterdam

May 27th, 2013

Goals of the course

Discuss recent examples of:

Goals of the course

Discuss recent examples of:

1. Logics motivated by CogSci;

Goals of the course

Discuss recent examples of:

1. Logics motivated by CogSci;
2. Experiments motivated by logics;

Goals of the course

Discuss recent examples of:

1. Logics motivated by CogSci;
2. Experiments motivated by logics;
3. Selection of experimental methods.

How can logic contribute?

1. In building cognitive theories;

How can logic contribute?

1. In building cognitive theories;
2. In computational modeling;

How can logic contribute?

1. In building cognitive theories;
2. In computational modeling;
3. In designing experiments.

Evaluating cognitive models

Along the following dimensions:

- logical relationships, e.g., incompatibility or identity
- explanatory power;
- computational plausibility.

Outline

Outline

Divide between logic and psychology

- Antiquity: a tool for analyzing human reasoning;

Divide between logic and psychology

- Antiquity: a tool for analyzing human reasoning;
- 19th century:
    - Beginnings of modern logic
    - Beginnings of modern psychology

Divide between logic and psychology

- Antiquity: a tool for analyzing human reasoning;
- 19th century:
  - Beginnings of modern logic
  - Beginnings of modern psychology
- Frege's anti-psychologism enforced separation

Divide between logic and psychology

- Antiquity: a tool for analyzing human reasoning;
- 19th century:
  - Beginnings of modern logic
  - Beginnings of modern psychology
- Frege's anti-psychologism enforced separation
- '60 witness the growth of cognitive science

Information processing and 3 levels of Marr

1. Computational level:
   - specify cognitive task:
     - f: initial state $\longrightarrow$ desired state
   - problems that a cognitive ability has to overcome

Information processing and 3 levels of Marr

1. Computational level:
   - specify cognitive task:
     - f: initial state $\longrightarrow$ desired state
   - problems that a cognitive ability has to overcome
2. Algorithmic level:
   - the algorithms that may be used to achieve a solution
   - compute f

Information processing and 3 levels of Marr

1. Computational level:
   - specify cognitive task:
     - f: initial state $\longrightarrow$ desired state
   - problems that a cognitive ability has to overcome
2. Algorithmic level:
   - the algorithms that may be used to achieve a solution
   - compute f
3. Implementation level:
   - how this is actually done in neural activity

Marr, Vision: a computational investigation into the human representation and processing visual information, 1983

Extending Classical Perspective

Extending Classical Perspective

Observation
Logical analysis informs about intrinsic properties of a problem.

Extending Classical Perspective

Observation

Logical analysis informs about intrinsic properties of a problem.

$\hookrightarrow$ Level 1.5

Outline

Outline

Syllogisms

Example

No climbers are weak.
Some climbers are logicians.

Some logicians aren't weak.

Monotonicity

↑MON	$Q_M[A, B]$ and $A \subseteq A' \subseteq M$ then $Q_M[A', B]$.
↓MON	$Q_M[A, B]$ and $A' \subseteq A \subseteq M$ then $Q_M[A', B]$.
MON↑	$Q_M[A, B]$ and $B \subseteq B' \subseteq M$ then $Q_M[A, B']$.
MON↓	$Q_M[A, B]$ and $B' \subseteq B \subseteq M$ then $Q_M[A, B']$.

Square of opposition

some is $\uparrow MON \uparrow$

no is $\downarrow MON \downarrow$

not all is $\uparrow MON \downarrow$

all is $\downarrow MON \uparrow$

Reasoning Patterns

Definition
If Q is increasing and $X \subseteq Y$, then Q(A, X) entails Q(A, Y).

1. Every boy runs fast.
2. Every boy runs.

Definition
If Q is decreasing and $Y \subseteq X$, then Q(A, X) entails Q(A, Y).

1. No boy runs.
2. No boy runs fast.

Classical Syllogistic

- All A are B : universal affirmative (A)
- Some A are B: particular affirmative (I)
- No A are B: universal negative (E)
- Some A are not B: particular negative (O)

Figure 1	*Figure 2*	*Figure 3*	*Figure 4*
B C	C B	B C	C B
A B	A B	B A	B A
A C	A C	A C	A C

$$\text{AE4O:}\quad \frac{\begin{array}{l}\text{All C are B}\\ \text{No B are A}\end{array}}{\text{Some A are not C}}$$

Syllogistic Reasoning: Meta-data Analysis

Table 1
Percentage of times each syllogistic conclusion was endorsed according to the meta-analysis by Chater and Oaksford (1999)[a]

premisses & figure	*conclusion* A	I	E	O	*premisses & figure*	*conclusion* A	I	E	O	*premisses & figure*	*conclusion* A	I	E	O
AA1	90	5	0	0	AO1	1	6	1	57	IO1	3	4	1	30
AA2	58	8	1	1	AO2	0	6	3	67	IO2	1	5	4	37
AA3	57	29	0	0	AO3	0	10	0	66	IO3	0	9	1	29
AA4	75	16	1	1	AO4	0	5	3	72	IO4	0	5	1	44
AI1	0	92	3	3	OA1	0	3	3	68	OI1	4	6	0	35
AI2	0	57	3	11	OA2	0	11	5	56	OI2	0	8	3	35
AI3	1	89	1	3	OA3	0	15	3	69	OI3	1	9	1	31
AI4	0	71	0	1	OA4	1	3	6	27	OI4	3	8	2	29
IA1	0	72	0	6	II1	0	41	3	4	EE1	0	1	34	1
IA2	13	49	3	12	II2	1	42	3	3	EE2	3	3	14	3
IA3	2	85	1	4	II3	0	24	3	1	EE3	0	0	18	3
IA4	0	91	1	1	II4	0	42	0	1	EE4	0	3	31	1
AE1	0	3	59	6	IE1	1	1	22	16	EO1	1	8	8	23
AE2	0	0	88	1	IE2	0	0	39	30	EO2	0	13	7	11
AE3	0	1	61	13	IE3	0	1	30	33	EO3	0	0	9	28
AE4	0	3	87	2	IE4	0	1	28	44	EO4	0	5	8	12
EA1	0	1	87	3	EI1	0	5	15	66	OE1	1	0	14	5
EA2	0	0	89	3	EI2	1	1	21	52	OE2	0	8	11	16
EA3	0	0	64	22	EI3	0	6	15	48	OE3	0	5	12	18
EA4	1	3	61	8	EI4	0	2	32	27	OE4	0	19	9	14
										OO1	1	8	1	22
										OO2	0	16	5	10
										OO3	1	6	0	15
										OO4	1	4	1	25

A = all E = no
I = some O = some … not

[a] All figures have been rounded to the nearest integer; valid conclusions are shaded. Whenever two conclusions in the same row are valid, only the first one is valid in predicate logic.

Chater and Oaksford, The Probability Heuristic Model of Syllogistic Reasoning, Cognitive Psychology, 1999

Monotonicity Calculus

- Logic rendering many valid arguments.
- Including syllogistic.
- Pivoting on monotonicity, e.g.,

Monotonicity Calculus

- Logic rendering many valid arguments.
- Including syllogistic.
- Pivoting on monotonicity, e.g.,

Rule 1	Rule 2	Example 1	Example 2
$\alpha \implies \beta$	$\beta \implies \alpha$	*all*(A, B)	*all*(C, B)
$\dots \alpha^{+} \dots$	$\dots \alpha^{-} \dots$	*some*(A^{+}, C)	*no*(B^{-}, A)
$\dots \beta^{+} \dots$	$\dots \beta^{-} \dots$	*some*(B^{+}, C)	*no*(C^{-}, A)

Monotonicity Calculus

- Logic rendering many valid arguments.
- Including syllogistic.
- Pivoting on monotonicity, e.g.,

$$\frac{\alpha \implies \beta \quad \ldots\alpha^{+}\ldots}{\ldots\beta^{+}\ldots}\ \text{(Rule 1)} \qquad \frac{\beta \implies \alpha \quad \ldots\alpha^{-}\ldots}{\ldots\beta^{-}\ldots}\ \text{(Rule 2)} \qquad \frac{\textit{all}(A, B) \quad \textit{some}(A^{+}, C)}{\textit{some}(B^{+}, C)}\ \text{(Example 1)} \qquad \frac{\textit{all}(C, B) \quad \textit{no}(B^{-}, A)}{\textit{no}(C^{-}, A)}\ \text{(Example 2)}$$

$$\frac{\mathrm{Q}(A, B)}{\mathrm{Q}(B, A),\ \mathrm{Q} = \textit{some}}\ \text{(Conversion)} \qquad \frac{\mathrm{no}(\mathrm{A}, \mathrm{B})}{\textit{all}(A, \textit{not } B)}\ \text{(No/All-not)}$$

Processing Model: An Example

Example

1. $no(B, C)$ premiss
2. $some(B, A)$ premiss

Processing Model: An Example

Example

1. $no(B, C)$ premiss
2. $some(B, A)$ premiss
3. $some(A, B^{+})$ Conv from 2

Processing Model: An Example

Example

1. $no(B, C)$ premiss
2. $some(B, A)$ premiss
3. $some(A, B^{+})$ Conv from 2
4. $all(B, not\ C)$ No/All-not from 1

Processing Model: An Example

Example

1. $no(B, C)$ premiss
2. $some(B, A)$ premiss
3. $some(A, B^+)$ Conv from 2
4. $all(B, not\ C)$ No/All-not from 1
5. $some(A, not\ C)$ Mon from 3 and 4

Processing Model: An Example

1. The shorter the proof the easier the syllogism.

Processing Model: An Example

1. The shorter the proof the easier the syllogism.
   - Level 1.5,

Processing Model: An Example

1. The shorter the proof the easier the syllogism.
   - Level 1.5,
   - Rule applications may be empirically weighted,
   - 100-(20 for Rule1, 2; 10 for No/All-not)

Processing Model: An Example

1. The shorter the proof the easier the syllogism.
   - Level 1.5,
   - Rule applications may be empirically weighted,
   - 100-(20 for Rule1, 2; 10 for No/All-not)
2. It gives a good fit with data.

Table 4
Predicted difficulty of valid syllogisms according to the model described in the text, compared with Chater and Oaksford's scores (in parentheses)

AA1A	80	(90)	OA3O	70	(69)	EA1O	40	(3)
EA1E	80	(87)	AO2O	70	(67)	EA2O	40	(3)
EA2E	80	(89)	EI1O	60	(66)	EA3O	40	(22)
AE2E	80	(88)	EI2O	60	(52)	EA4O	40	(8)
AE4E	80	(87)	EI3O	60	(48)	AE2O	40	(1)
IA3I	80	(85)	EI4O	60	(27)	AE4O	40	(2)
IA4I	80	(91)	AA1I	60	(5)			
AI1I	80	(92)	AA3I	60	(29)			
AI3I	80	(89)	AA4I	60	(16)			

Geurts, Reasoning with Quantifiers, Cognition, 2003

Monotonicity Profiles Determine Difficulty

1. Some of the sopranos sang with more than three of the tenors.
2. None of the sopranos sang with fewer than three of the tenors.
3. Some of the sopranos sang with fewer than three of the tenors.

Monotonicity Profiles Determine Difficulty

1. Some of the sopranos sang with more than three of the tenors.
2. None of the sopranos sang with fewer than three of the tenors.
3. Some of the sopranos sang with fewer than three of the tenors.

$$\frac{\begin{array}{c}\mathrm{Q}_1 A \text{ sang with } \mathrm{Q}_2 B \\ \text{All B were C./All C were B.}\end{array}}{\mathrm{Q}_1 A \text{ sang with } \mathrm{Q}_2 C}$$

Monotonicity Profiles Determine Difficulty

1. Some of the sopranos sang with more than three of the tenors.
2. None of the sopranos sang with fewer than three of the tenors.
3. Some of the sopranos sang with fewer than three of the tenors.

$$\frac{\begin{array}{c}\mathrm{Q}_1\,A \text{ sang with } \mathrm{Q}_2\,B\\ \text{All B were C./All C were B.}\end{array}}{\mathrm{Q}_1\,A \text{ sang with } \mathrm{Q}_2\,C}$$

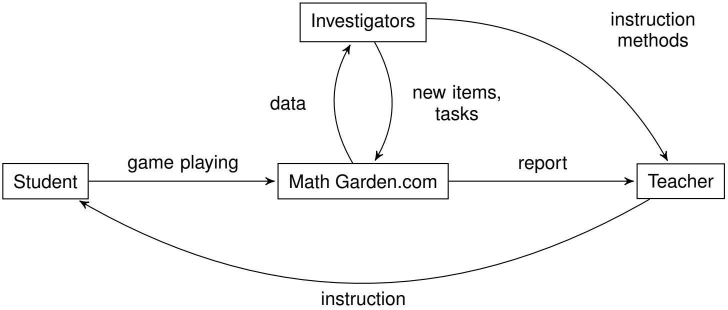
$$\uparrow \mathrm{Q}_1 \uparrow \mathrm{Q}_2 < \begin{array}{c}\uparrow \mathrm{Q}_1 \downarrow \mathrm{Q}_2\\ \downarrow \mathrm{Q}_1 \uparrow \mathrm{Q}_2\\ \downarrow \mathrm{Q}_1 \downarrow \mathrm{Q}_2\end{array} \qquad \uparrow \mathrm{Q}_1 \uparrow \mathrm{Q}_2 < \downarrow \mathrm{Q}_1 \downarrow \mathrm{Q}_2 < \begin{array}{c}\uparrow \mathrm{Q}_1 \downarrow \mathrm{Q}_2\\ \downarrow \mathrm{Q}_1 \uparrow \mathrm{Q}_2\end{array}$$

Geurts and Van der Slik, Monotonicity and Processing Load, Journal of Semantics, 2005

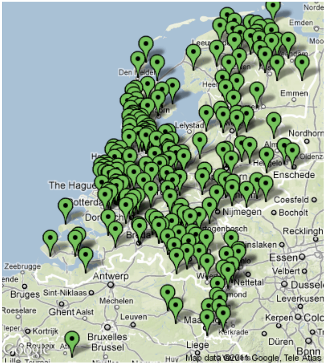
Outline

Math Garden (Rekentuin.nl or MathsGarden.com)

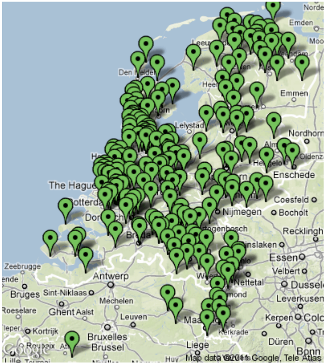
- adaptive training environment with educational games
- abstract thinking development
- 15 arithmetic games and 2 complex reasoning games



Math Garden (Rekentuin.nl or MathsGarden.com)



Math Garden (Rekentuin.nl or MathsGarden.com)



- In three years the number of schools has grown from 8 to over 700.
- Currently, over 60,000 students have generated over 150 billion.

Difficulty Levels

- students play game-items suited for their level (70% correctly)
- the tasks' difficulty and the students' level are continuously estimated
- via the Elo rating system (used for ranking chess players, Elo 1978)
- i.e.: students are ranked by playing, and items are rated by getting played
- ratings depend on accuracy and speed of item solving (Klinkenberg 2011)

By-products:

1) rating of all items (item difficulty parameters)
2) rating of children (reflecting the reasoning ability)

The Goal

introduce a dedicated logical reasoning training in primary schools

understand the empirically established item difficulty parameters

by means of a logical analysis of the items

The Goal

introduce a dedicated logical reasoning training in primary schools

understand the empirically established item difficulty parameters

by means of a logical analysis of the items

computational paradigms for cognition

Mastermind Game

- Meirowitz 1970, but similar to the earlier Bulls and Cows
- an inductive inquiry game, trials of experimentation and evaluation

Mastermind Game

- Meirowitz 1970, but similar to the earlier Bulls and Cows
- an inductive inquiry game, trials of experimentation and evaluation

Mastermind Game: A Code-Breaking Game

- the set consists of:
  - a decoding board
  - code pegs of k colors
  - and feedback pegs of black and white
- players:
  - the code-maker: chooses a secret pattern of ℓ code pegs
  - the code-breaker: guesses the pattern, in a given n rounds
- rounds:
  - code-breaker makes a guess by placing a row of ℓ code pegs
  - code-maker provides the feedback:
    one black for each code peg of correct color and position, and
    one white for each peg of correct color but wrong position
  - repeat until either the code-breaker guesses correctly, or n incorrect guesses have been made
- winning:
  - for the code-breaker: if obtains the solution within n rounds
  - the code-maker wins otherwise

Mastermind: A Logical Game

- question of the underlying logical reasoning and its difficulty
- mathematical results only on existence of efficient strategies (Knuth 1976, Irving 1978, Koyama 1993, Kooi 2005)

Static Mastermind (Chvatal 1983)

Static Mastermind is a version of the Mastermind game

Static Mastermind (Chvatal 1983)

Static Mastermind is a version of the Mastermind game

- the goal is to find out the minimum number of guesses
- the code-breaker can make all at once
- at the beginning of the game
- without waiting for the individual feedbacks

Static Mastermind (Chvatal 1983)

Static Mastermind is a version of the Mastermind game

- the goal is to find out the minimum number of guesses
- the code-breaker can make all at once
- at the beginning of the game
- without waiting for the individual feedbacks
- and upon receiving them all at once
- completely determine the code in the next guess

Complexity of Static Mastermind (Stuckman 2006)

.

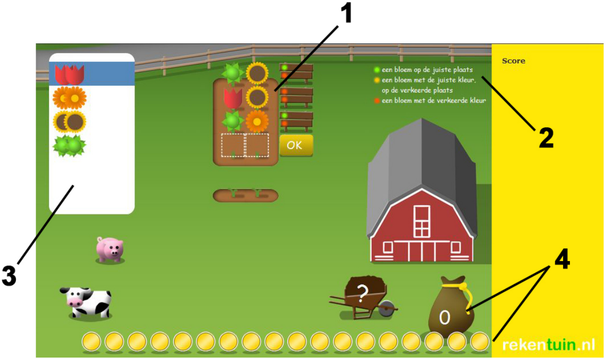
Definition (Mastermind Satisfiability Decision Problem)

Input	A set of guesses G and their corresponding scores.
Question	Is there at least one valid solution?

Theorem

Mastermind Satisfiability Decision Problem is NP-complete wrt ℓ (positions).

Deductive Mastermind: Flowercode in Math Garden



1) decoding board
2) short feedback instruction
3) domain of flowers to choose from
4) timer in the form of disappearing coins

Some Facts about Deductive Mastermind

- unlike the classic, DM does not have the trial conjectures
- DM gives the clues upfront
- unlike SM, ensuring exactly one correct solution

reduced complexity: from inductive inference to logical-reasoning game

DM is a combo of the classic (the goal is the same) and SM (no trial-and-error)

emphasis on the atomic logical steps of non-linguistic logical reasoning

DM is easily adaptable as a single-player game (Math Garden)

Some Facts about Flowercode

- running since November 2010

Some Facts about Flowercode

- running since November 2010
- 321 game-items, 1-5 flowers, 2-5 colors

Some Facts about Flowercode

- running since November 2010
- 321 game-items, 1-5 flowers, 2-5 colors
- by December 2012, 4,895,648 items had been played

Some Facts about Flowercode

- running since November 2010
- 321 game-items, 1-5 flowers, 2-5 colors
- by December 2012, 4,895,648 items had been played
- by 37,339 primary school students (grades 1-6, age: 6-12 years)

Some Facts about Flowercode

- running since November 2010
- 321 game-items, 1-5 flowers, 2-5 colors
- by December 2012, 4,895,648 items had been played
- by 37,339 primary school students (grades 1-6, age: 6-12 years)
- in over 700 locations (schools and family homes)

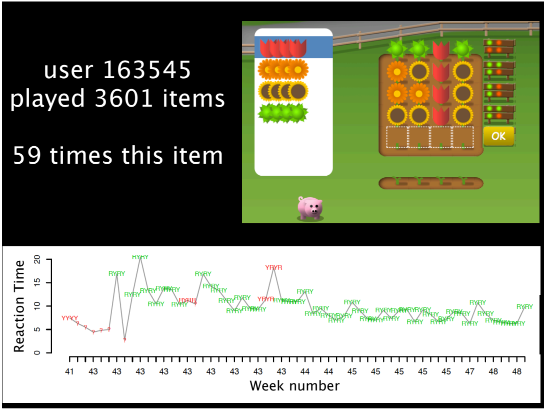
Some Facts about Flowercode

- running since November 2010
- 321 game-items, 1-5 flowers, 2-5 colors
- by December 2012, 4,895,648 items had been played
- by 37,339 primary school students (grades 1-6, age: 6-12 years)
- in over 700 locations (schools and family homes)

We can access:

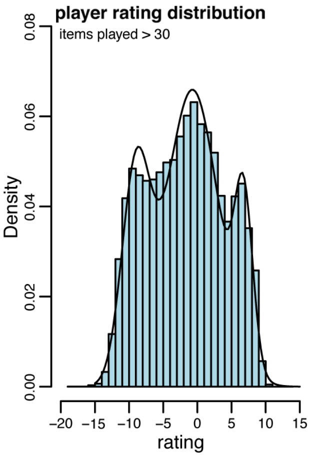
- the individual progress of individual players on a single game
- the most frequent mistakes with respect to a game-item
- the relative difficulty of game-items
- correlations with different, mathematical games
- etc.

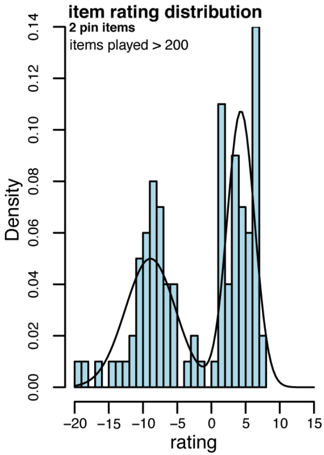
Some Facts about Flowercode



The Necessity of Prior Difficulty Assessment

initial difficulty estimation in terms of non-logical aspects
(number of flowers, colors, lines, the rate of the hypotheses elimination, etc.)





how to fix this to facilitate the training effect?

A Logical Analysis: Conjectures

Each Deductive Mastermind game-item consists of a sequence of conjectures.

Definition

A conjecture of length l over k colors is any sequence given by a total assignment, $h : \{1, \ldots, \ell\} \to \{c_1, \ldots, c_k\}$. The *goal sequence* is a distinguished conjecture, $goal : \{1, \ldots, \ell\} \to \{c_1, \ldots, c_k\}$.

A Logical Analysis: Feedback

- every non-goal conjecture is accompanied by a feedback
- that indicates how similar *h* is to the given goal assignment
- feedback colors g, o, r

Definition

Let *h* be a conjecture and let *goal* be the goal sequence, both of length *l* over *k* colors. The feedback *f* for *h* with respect to *goal* is a sequence

$$\overbrace{g \dots g}^{a} \overbrace{o \dots o}^{b} \overbrace{r \dots r}^{c} = g^a o^b r^c,$$

where $a, b, c \in \mathbb{N}$ and $a + b + c = \ell$.

The feedback consists of:

- exactly one g for each $i \in G$, where $G = \{i \in \{1, \dots \ell\} \mid h(i) = goal(i)\}$.
- exactly one o for every $i \in O$, where $O = \{i \in \{1, \dots, \ell\} \backslash G \mid \exists j \in \{1, \dots, \ell\} \backslash G,$ such that $i \neq j$ and $h(i) = goal(j)\}$.
- exactly one r for every $i \in \{1, \dots, \ell\} \backslash (G \cup O)$.

The informational content of the feedback

a second-order formula that encodes any feedback $g^a o^b r^c$ for any h wrt *goal*

$$
\begin{aligned}
&\exists G \subseteq \{1, \dots \ell\}(card(G)=a \land \forall i{\in}G\ h(i)=goal(i) \land \forall i{\notin}G\ h(i)\neq goal(i) \\
&\land \exists O \subseteq \{1, \dots \ell\} \setminus G\ (card(O)=b \land \forall i{\in}O\ \exists j{\in}\{1, \dots \ell\} \setminus G(j{\neq}i \land h(i)=goal(j)) \\
&\land \forall i{\in}\{1, \dots \ell\} \setminus (G \cup O)\ \forall j{\in}\{1, \dots \ell\} \setminus G\ h(i)\neq goal(j))).
\end{aligned}
$$

The informational content of the feedback

a general method of providing a propositional formula for any (h, f)

- literals: $h(i) = goal(j)$, where $i, j \in \{1, \dots \ell\}$ (or $p_{i,j}$, for $i, j \in \{1, \dots \ell\}$)
- $\varphi^g_G, \varphi^o_{G,O}, \varphi^r_{G,O}$ correspond to different parts of feedback:
  - $\varphi^g_G := \bigwedge_{i \in G} h(i){=}goal(i) \wedge \bigwedge_{j \in \{1,\dots,\ell\}-G} h(j){\neq}goal(j)$
  - $\varphi^o_{G,O} := \bigwedge_{i \in O}(\bigvee_{j \in \{1,\dots,\ell\}-G, i \neq j} h(i) = goal(j))$
  - $\varphi^r_{G,O} := \bigwedge_{i \in \{1,\dots\ell\} \setminus (G \cup O), j \in \{1,\dots\ell\} \setminus G, i \neq j} h(i) \neq goal(j)$

as many substitutions of the above as choices of sets G and O

The informational content of the feedback

- set $\mathbb{G} := \{G | G \subseteq \{1, \ldots, \ell\} \wedge card(G) = a\}$, and,
- if $G \subseteq \{1, \ldots, \ell\}$, then $\mathbb{O}^G = \{O | O \subseteq \{1, \ldots, \ell\} \setminus G \wedge card(O) = b\}$

Definition

Finally, we can set $Bt(h, f)$, the Boolean translation of (h, f) to be given by:

$$Bt(h, f) := \bigvee_{G \in \mathbb{G}} (\varphi^g_G \wedge \bigvee_{O \in \mathbb{O}^G} (\varphi^o_{G,O} \wedge \varphi^r_{G,O})).$$

An Example

Example

Let us take $\ell = 2$ and (h, f) such that: $h(1){:=}c_1$, $h(2){:=}c_2$; $f{:=}or$. Then $\mathbb{G}{=}\{\emptyset\}$, $\mathbb{O}^{\{\emptyset\}}{=}\{\{1\},\{2\}\}$. The corresponding formula, $Bt(h, f)$, is:

$(goal(1){\neq}c_1 \wedge goal(2){\neq}c_2) \wedge ((goal(1){=}c_2 \wedge goal(2){\neq}c_1) \vee (goal(2){=}c_1 \wedge goal(1){\neq}c_2))$

DM Game-item

Definition

A Deductive Mastermind game-item over ℓ positions, k colors and n lines, $DM(l, k, n)$, is a set $\{(h_1, f_1), \ldots, (h_n, f_n)\}$ of pairs, each consisting of a single conjecture together with its corresponding feedback. Respectively, $Bt(DM(l, k, n)) = Bt(\{(h_1, f_1), \ldots, (h_n, f_n)\}) = \{Bt(h_1, f_1), \ldots, Bt(h_n, f_n)\}$.

- hence, each DM game-item is a set of Boolean formulae
- moreover, by the construction this set is satisfiable
- and, even more, there is a unique valuation

Analytic Tableaux for Deductive Mastermind

- analytic tableau is a decision procedure for propositional logic
- it solves satisfiability of finite sets of formulas of propositional logic
- by giving an adequate valuation

Analytic Tableaux for Deductive Mastermind

- analytic tableau is a decision procedure for propositional logic
- it solves satisfiability of finite sets of formulas of propositional logic
- by giving an adequate valuation
- building a formula-labeled tree rooted at the set
- unfolding breaks them into smaller formulae
- until contradiction is found or no further reduction is possible

Analytic Tableaux for Deductive Mastermind

- analytic tableau is a decision procedure for propositional logic
- it solves satisfiability of finite sets of formulas of propositional logic
- by giving an adequate valuation
- building a formula-labeled tree rooted at the set
- unfolding breaks them into smaller formulae
- until contradiction is found or no further reduction is possible

$$
\begin{tikzcd}
\varphi \wedge \psi \arrow[d, dash, "\wedge"] \\
\varphi, \psi
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
 & \varphi \vee \psi \arrow[dl, dash] \arrow[dr, dash] \arrow[d, phantom, "\vee"] & \\
\varphi & & \psi
\end{tikzcd}
$$

Analytic Tableau and DM: An Observation

By construction of DM

Applying the analytic tableaux method to the Boolean translation of a Deductive Mastermind game-item will give the unique missing assignment *goal*.

2-placed Deductive Mastermind Game-items

2-placed Deductive Mastermind Game-items

$$gg, go, oo, rr, gr, or$$

$$c_i, c_j$$

$$\Big|\ oo$$

$$goal(1) \neq c_i$$
$$goal(2) \neq c_j$$
$$goal(1) = c_j$$
$$goal(2) = c_i$$

2-placed Deductive Mastermind Game-items

$$gg, go, oo, rr, gr, or$$

$$\begin{tikzcd} c_i, c_j \arrow[d, dash, "oo"] \\ \begin{array}{l} goal(1) \neq c_i \\ goal(2) \neq c_j \\ goal(1) = c_j \\ goal(2) = c_i \end{array} \end{tikzcd} \qquad \begin{tikzcd} c_i, c_j \arrow[d, dash, "rr"] \\ \begin{array}{ll} goal(1) \neq c_i & goal(2) \neq c_j \\ goal(1) \neq c_j & goal(2) \neq c_i \end{array} \end{tikzcd}$$

2-placed Deductive Mastermind Game-items

$$gg, go, \color{red}{oo}, \color{red}{rr}, \color{red}{gr}, \color{red}{or}$$

c_i, c_j

oo

$goal(1) \neq c_i$
$goal(2) \neq c_j$
$goal(1) = c_j$
$goal(2) = c_i$

c_i, c_j

rr

$goal(1) \neq c_i \quad goal(2) \neq c_j$
$goal(1) \neq c_j \quad goal(2) \neq c_i$

c_i, c_j

gr

$goal(1) = c_i \quad goal(2) = c_j$
$goal(2) \neq c_j \quad goal(1) \neq c_i$

2-placed Deductive Mastermind Game-items

$$gg, go, oo, rr, gr, or$$

c_i, c_j

oo

$goal(1) \neq c_i$
$goal(2) \neq c_j$
$goal(1) = c_j$
$goal(2) = c_i$

c_i, c_j

rr

$goal(1) \neq c_i$ $goal(2) \neq c_j$
$goal(1) \neq c_j$ $goal(2) \neq c_i$

c_i, c_j

gr

$goal(1) = c_i$
$goal(2) \neq c_j$

$goal(2) = c_j$
$goal(1) \neq c_i$

c_i, c_j

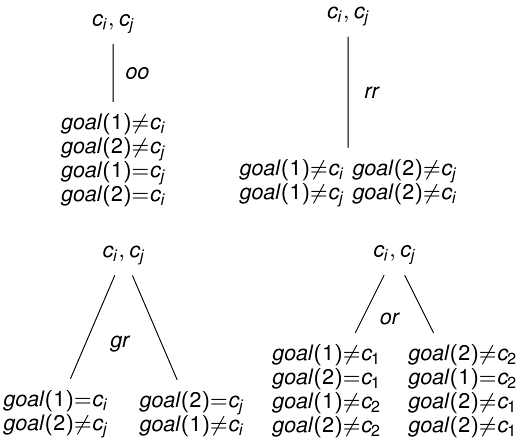
or

$goal(1) \neq c_1$
$goal(2) = c_1$
$goal(1) \neq c_2$
$goal(2) \neq c_2$

$goal(2) \neq c_2$
$goal(1) = c_2$
$goal(2) \neq c_1$
$goal(2) \neq c_1$

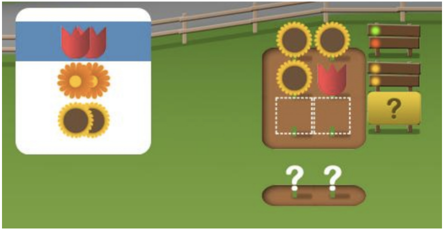
2-placed Deductive Mastermind Game-items

gg, go, oo, rr, gr, or

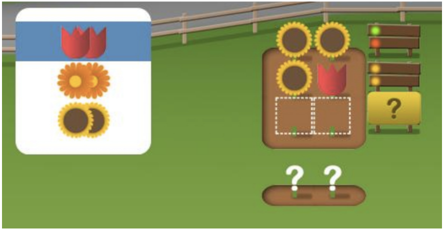


$oo < rr < gr < or$

An Example



An Example

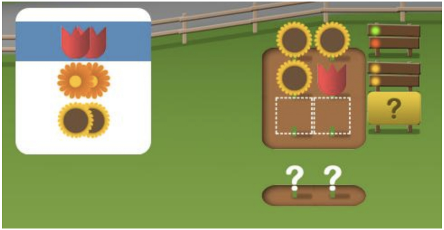


$Bt(h_1, f_1)$
$Bt(h_2, f_2)$

gr

$goal(1) = c_1$	$goal(2) = c_1$
$goal(2) \neq c_1$	$goal(1) \neq c_1$
$Bt(h_2, f_2)$	$Bt(h_2, f_2)$
oo	oo
$goal(1) = c_2$	$goal(1) = c_2$
$goal(2) = c_1$	$goal(2) = c_1$

An Example



$$\begin{array}{c} Bt(h_1, f_1) \\ Bt(h_2, f_2) \end{array}$$

gr

$$\begin{array}{c} goal(1)=c_1 \\ goal(2)\neq c_1 \\ Bt(h_2, f_2) \end{array} \qquad \begin{array}{c} goal(2)=c_1 \\ goal(1)\neq c_1 \\ Bt(h_2, f_2) \end{array}$$

oo $\qquad$ oo

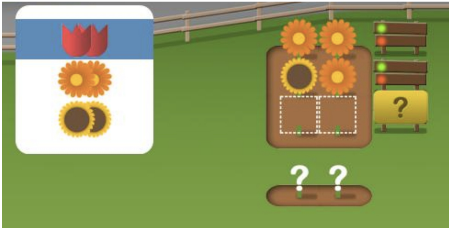
$$\begin{array}{c} goal(1)=c_2 \\ goal(2)=c_1 \end{array} \qquad \begin{array}{c} goal(1)=c_2 \\ goal(2)=c_1 \end{array}$$

$$\begin{array}{c} Bt(h_2, f_2) \\ Bt(h_1, f_1) \end{array}$$

oo

$$\begin{array}{c} goal(1)=c_2 \\ goal(2)=c_1 \\ Bt(h_1, f_1) \end{array}$$

Another Example



$Bt(h_1, f_1)$
$Bt(h_2, f_2)$

$goal(1) = c_1$ $\quad$ gr $\quad$ $goal(2) = c_1$
$goal(2) \neq c_1$ $\quad$ $goal(1) \neq c_1$
$Bt(h_2, f_2)$ $\quad$ $Bt(h_2, f_2)$

gr $\quad$ gr

$goal(1) = c_2$ $\quad$ $goal(2) = c_1$ $\quad$ $goal(1) = c_2$ $\quad$ $goal(2) = c_1$
$goal(2) \neq c_1$ $\quad$ $goal(1) \neq c_2$ $\quad$ $goal(2) \neq c_1$ $\quad$ $goal(1) \neq c_2$

Hypotheses and Preliminary Results

- tableau give ‘ideal’ reasoning scheme
- abstract complexity measure (tree size)
- shape and size of the tree depends on what goes first (minimal size)
- reasoning optimization

items’ initial difficulty corresponds to the size of top-botom trees

items’ logical difficulty corresponds to the size of the minimal trees

the reasoning is optimized according to feedback complexity

Method

- participants: 28,247 students from grades 1-6, of age: 6-12 years.
- played: 2,187,354 items between November 2010 and January 2012
- items: 321 DM items among them 100 two-places items.

Results

all factors but one (*gr*) were significant in predicting item difficulties

Results

all factors but one (*gr*) were significant in predicting item difficulties

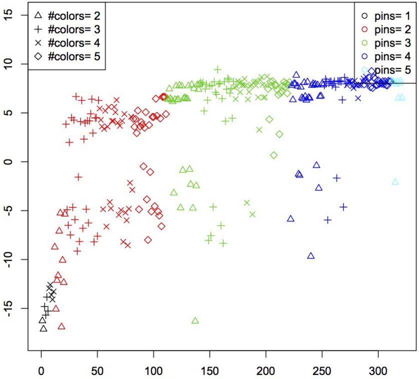
two difficulty clusters

1) easy

   no *or* feedback and no *gr* feedback

   no *or* feedback, at least one *gr* feedback, and all colors are included

2) difficult: otherwise



Conclusions and Further Work

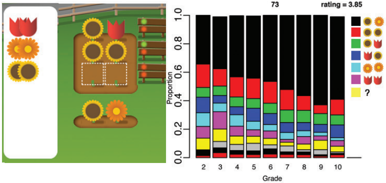
- using proof-theory to analyze the cognitive difficulty of logical reasoning
- reasonably good prediction of item-difficulties
- other factors: the motor requirements (number of clicks)

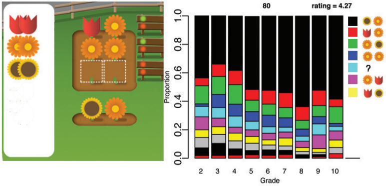
Further work:

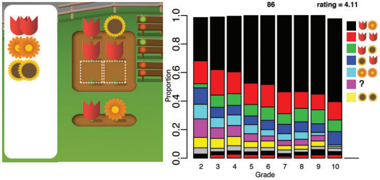
theoretical various complexity measures based on tableau
extending the logical insights to other games, like Sudoku

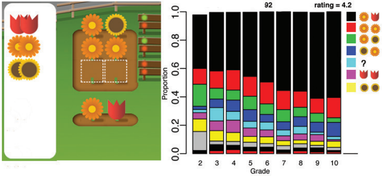
empirical extend to arbitrary conjectures
compare individual abilities in DM with other games

Error Analysis









Learning and Interaction

THE END OF LECTURE 1