

# Recommender Systems and Education

Radek Pelánek

2014

# Technology and Education

e-learning, m-learning, intelligent tutoring system, technology-enhanced learning, computer-based instruction, computer managed instruction, computer-based training, computer-assisted instruction, computer-aided instruction, internet-based training, flexible learning, web-based training, online education, massive open online courses, virtual education, virtual learning environments, digital education, multimedia learning, ...

# This Lecture

- relation to topics discussed so far
- focus on
  - specific examples
  - personalization and different types of recommendations

<i>Tasks</i>	<i>Description</i>	<i>Generic recommender</i>	<i>TEL recommenders</i>	<i>New requirements</i>
<b>Existing User Tasks supported by Recommender Systems</b>				
1. ANNOTATION IN CONTEXT	Recommendations while user carries out other tasks	E.g. predicting how relevant the links are within a web page	E.g. predicting relevance/usefulness of items in the reading list of a course	Explore attributes for representing relevance/usefulness in a learning context
2. FIND GOOD ITEMS	Recommendations of suggested items	E.g. receiving list of web pages to visit	E.g. receiving a selected list of online educational resources around a topic	None
3. FIND ALL GOOD ITEMS	Recommendation of all relevant items	E.g. receiving a complete list of references on a topic	E.g. suggesting a complete list of scientific literature or blog postings around a topic	None
4. RECOMMEND SEQUENCE	Recommendation of a sequence of items	E.g. receive a proposed sequence of songs	E.g. receiving a proposed sequence through resources to achieve a particular learning goal	Explore formal and informal attributes for representing relevancy to a particular learning goal

5. JUST BROWSING	Recommendations out of the box while user is browsing	E.g. people that bought this, have also bought that	E.g. receiving recommendations for new courses on the university site	Explore formal and informal attributes for representing relevance/usefulness in a learning context
6. FIND CREDIBLE RECOMMENDER	Recommendations during initial exploration/testing phase of a system	E.g. movies that you will definitely like	E.g. restricting course recommendations to ones with high confidence /credibility	Explore criteria for measuring confidence and credibility in formal and informal learning

#### TEL User Tasks that could be supported by Recommender Systems

1. FIND NOVEL RESOURCES	Recommendations of particularly new or novel items	E.g. receiving recommendations about latest additions or particularly controversial items	E.g. receiving very new and/or controversial resources on covered topics	Explore recommendation techniques that select items beyond their similarity
2. FIND PEERS	Recommendation of other people with relevant interests	E.g. being suggested profiles of users with similar interests	E.g. being suggested peer students in the same class	Explore attributes for measuring the similarity with other people
3. FIND GOOD PATHWAYS	Recommendation of alternative learning paths through learning resources	E.g. receive alternative sequences of similar songs	E.g. receiving a list of alternative learning paths over the same resources to achieve a specific learning goal	Explore criteria for the construction and suggestion of alternative (but similar) sequences

<i>Name</i>	<i>Short description</i>	<i>Advantages</i>	<i>Disadvantages</i>	<i>Usefulness for TEL</i>
<i>Collaborative Filtering (CF) techniques</i>				
User-based CF	Users who rated the same item similarly probably have the same taste. Based on this assumption, this technique recommends the unseen items already rated by similar users.	No content analysis Domain-independent Quality improves Bottom-up approach Serendipity	New user problem New item problem Popular taste Scalability Sparsity Cold start problem	Benefit from experience Allocate learners to groups (based on similar ratings)
Item-based CF	Focus on items, assuming that the items rated similarly are probably similar. It recommends items with the highest correlation (based on ratings for the items).	No content analysis Domain-independent Quality improves Bottom-up approach Serendipity	New item problem Popular taste Sparsity Cold start problem	Benefit from experience
Stereotypes or demographics CF	Users with similar attributes are matched, then it recommends items that are preferred by similar users (based on user data instead of ratings).	No cold start problem Domain-independent Serendipity	Obtaining information Insufficient information Only popular taste Obtaining metadata information	Allocate learners to groups Benefit from experience Recommendation from the beginning of the PRS

Personal recommender systems for learners in lifelong learning networks: the requirements, techniques and model

### *Content-Based (CB) techniques*

Case-based reasoning	Assumes that if a user likes a certain item, s/he will probably also like similar items. Recommends new but similar items.	No content analysis Domain-independent Quality improves	New user problem Overspecialisation Sparsity Cold start problem	Keeps learner informed about learning goal Useful for hybrid RS
Attribute-based techniques	Recommends items based on the matching of their attributes to the user profile. Attributes could be weighted for their importance to the user.	No cold start problem No new user/new item problem Sensitive to changes of preferences Can include non-item-related features Can map from user needs to items	Does not learn with categories Ontology modelling and maintenance is required Overspecialisation	Useful for hybrid RS Recommendation from the beginning

Personal recommender systems for learners in lifelong learning networks: the requirements, techniques and model

# Education and RecSys

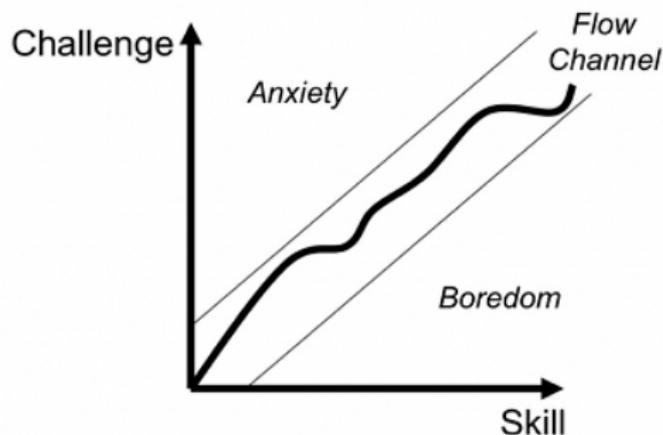
many techniques applicable in principle, but application more difficult than in “product recommendation”

- longer time frame
- pedagogical principles
- domain ontology, prerequisites
- learning outcomes not directly measurable (cf sales)

# Motivation: Personalization

- each student gets suitable learning materials, exercises
- tailored to a particular student, adequate for his knowledge
- mastery learning – fixed outcome, varied time  
(compared to classical education: fixed time, varied outcome)

# Motivation: Flow, ZPD



"Flow" concept by Mihaly Csikszentmihalyi. Drawn by Senia Maymin.

Vygotsky, zone of proximal development

# Adaptation and Personalization in Education

... gets lot of attention:

- Khan Academy
- MOOC courses
- Carnegie Learning
- Pearson
- ReasoningMind
- ...

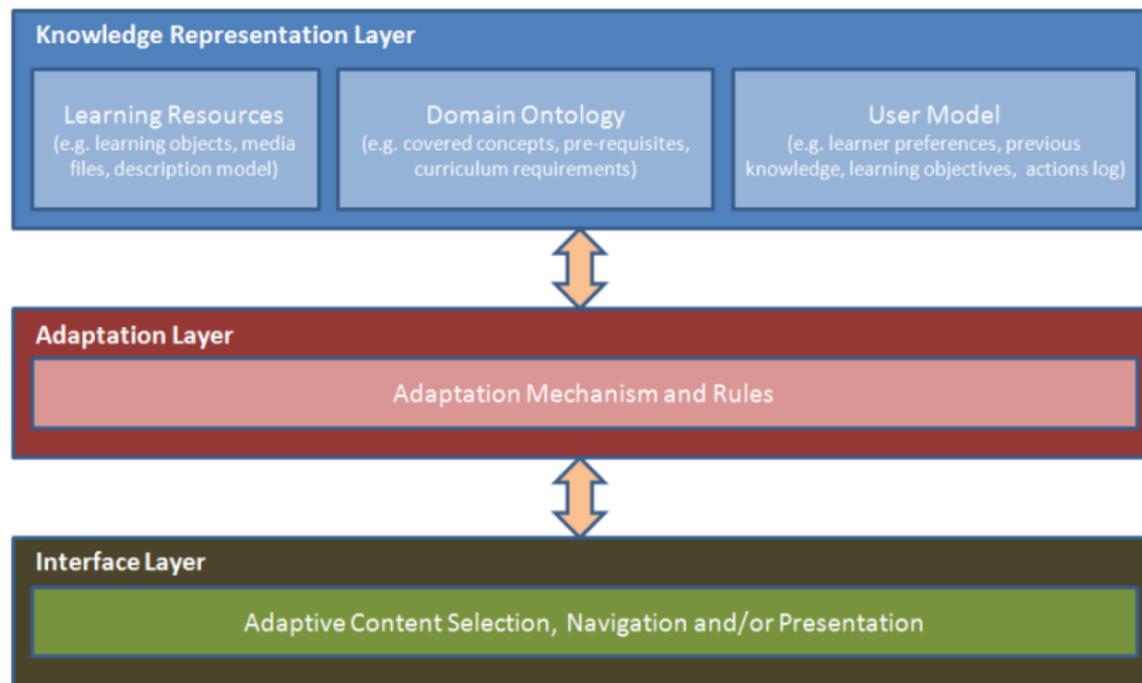
# Evaluation

- evaluation even more difficult than for other recommender systems
- compare goals:
  - product recommendations: sales
  - text (blogs, etc) recommendations: clicks (profit from advertisement)
  - education: learning
- learning can be measured only indirectly
- hard to tell what really works

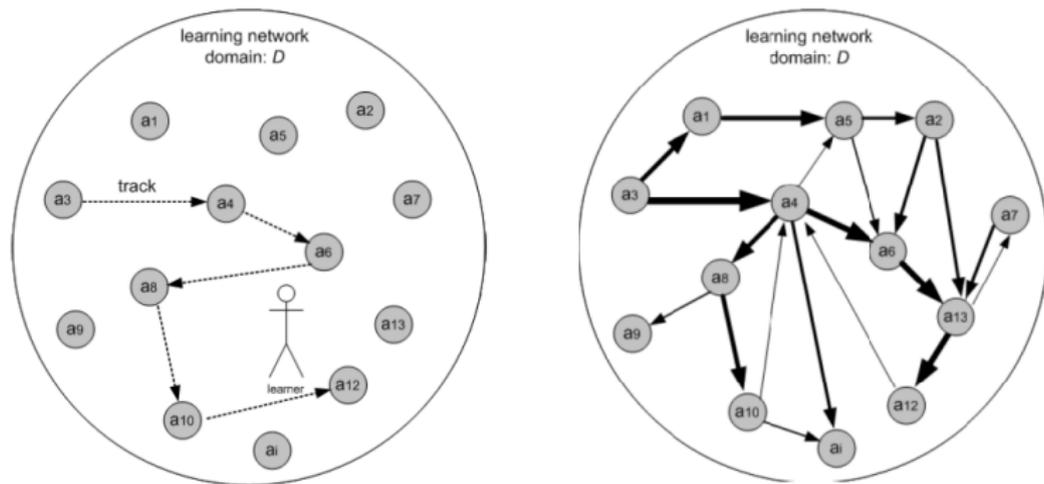
# Adaptive Educational Hypermedia

- adaptive content selection
  - most relevant items for particular user
- adaptive navigation support
  - navigation from one item to other
- adaptive presentation
  - presentation of the content

# Adaptive Educational Hypermedia



# Learning Networks



**Fig. 2.** Evolution of a learning network (left: starting phase with a first learner moving through possible learning activities; right: advanced phase showing emerging learning paths from the collective behavior of all learners)

# Intelligent Tutoring Systems

- behavior
  - outer loop – selection/recommendation of “items” (problems, exercises)
  - inner loop – hints, feedback, ...
- adaptation based on a student model
- knowledge modeling more involved than “taste modeling” (domain ontology, prerequisites, ...)

# Student Modeling and Collaborative Filtering

- user  $\sim$  student
- product  $\sim$  item, problem
- rating  $\sim$  student performance (correctness of answer, problem solving time, number of hints taken)

# Case Studies

- our projects (FI MU) – “adaptive practice”
  - Problem Solving Tutor
  - “Slepé mapy” – geography
  - “Umíme česky” – Czech grammar
- Wayang Outpost – math
- ALEF – programming
- CourseRank – course recommender

# Problem Solving Tutor

tutor.fi.muni.cz

- math and computer science problems, logic puzzles
- performance = problem solving time
- model – predictions of times
- recommendations – problems of similar difficulty

# Problem Solving Tutor

tutor.fi.muni.cz

PROBLEM SOLVING

TUTOR



Body: 4464



Jste přihášen jako **radek** v individuálním módu

Můj účet

přepnout na **Výukový mód**

Odhlásit

PROBLÉMY

STATISTIKY

VÝSLEDKOVKA

## Informatické



Interaktivní Python



Konečné automaty

```
#include <stdio.h>
#define N 9
main() {
    int x, y;
    printf(" ");
    for (x = 1; x <= N;
        printf("???", x);
        printf("\n\n");
}
```

Programování v C

pes zajíc  
kočka rys  
husa kozel

$^*[a-z](3,4)$$

Regulární výrazy



Robot Karel



Robotank

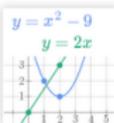


Želví grafika

## Matematické

	X	Y	Z
A	1	1	1
B	0	0	0
C	1	1	1

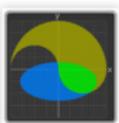
Binární křížovka



Grafář (nová verze)



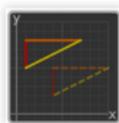
Matematické pexeso 2



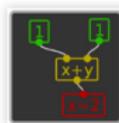
Obrazce



Rozbitá kalkulačka



Transformace



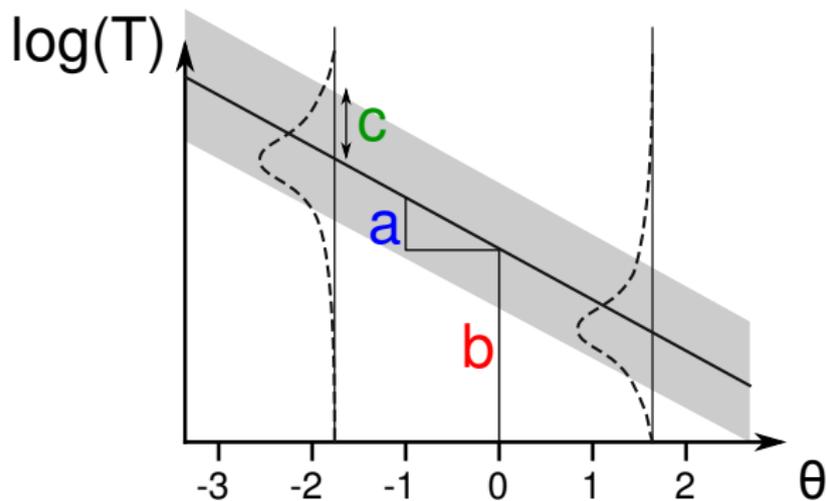
Výpočetní stromy

# Tutor: predictions

tutor.fi.muni.cz

<b>Kuželosečky - hyperboly</b> Neřešeno Předpověď 1:12	<b>Komplexní čísla - násobení</b> Vyřešeno Čas 0:58	<b>Logaritmy a mocniny - vzorečky</b> Neřešeno Předpověď 1:14	<b>Komplexní čísla - mocniny <math>i</math></b> Vyřešeno Čas 1:24	<b>Vlastnosti funkcí</b> Neřešeno Předpověď 1:17	<b>Kuželosečky 2</b> Vyřešeno Čas 1:05	<b>Zlomky</b> Neřešeno Předpověď 1:19	<b>Komplexní čísla - absolutní hodnoty</b> Vyřešeno Čas 0:45	<b>Logaritmy - hodnoty 2</b> Neřešeno Předpověď 1:23
<b>Kvadratické rovnice - řešení</b> Vyřešeno Čas 1:45	<b>Vzdálenosti</b> Vyřešeno Čas 1:16	<b>Kuželosečky</b> Neřešeno Předpověď 1:35	<b>Množiny - základní operace</b> Neřešeno Předpověď 1:42	<b>Kombinační čísla</b> Neřešeno Předpověď 1:42	<b>Kvadratická funkce 2</b> Neřešeno Předpověď 1:44	<b>Definiční obory a obory hodnot</b> Neřešeno Předpověď 1:46	<b>Logaritmy - vzorečky</b> Vyřešeno Čas 2:42	<b>Množiny</b> Neřešeno Předpověď 1:56
<b>Směs</b> Neřešeno Předpověď 2:02	<b>Derivace - goniometrické funkce</b> Vyřešeno Čas 2:15	<b>Součty</b> Vyřešeno Čas 1:10	<b>Kombinační čísla - vzorečky</b> Neřešeno Předpověď 2:17	<b>Komplexní čísla</b> Neřešeno Předpověď 2:17	<b>Úhly 2</b> Neřešeno Předpověď 2:29	<b>Nerovnosti</b> Neřešeno Předpověď 2:39	<b>Kuželosečky - kružnice</b> Neřešeno Předpověď 2:56	<b>Limity funkcí</b> Vyřešeno Čas 2:36

# Model of Problem Solving Times



# Parameter Estimation

- data: student  $s$  solved problem  $p$  in time  $t_{sp}$
- we need to estimate:
  - student skills  $\theta$
  - problem parameters  $a, b, c$
- stochastic gradient descent
- very similar to the “SVD” collaborative filtering algorithm

# Evaluation of Predictions

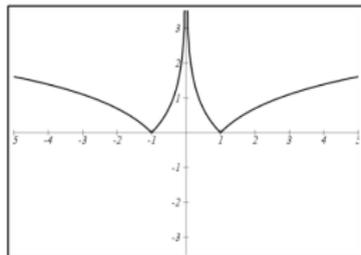
- 20 types of problems
- data: 5 000 users, 8 000 hours, more than 220 000 problems
- difficulty of problems: from 10 seconds to 1 hour
- train, test set
- metrics: RMSE
- results:
  - significant improvement with respect to a baseline (mean times)
  - more complex models do not bring much improvement

## stejná základní obtížnost

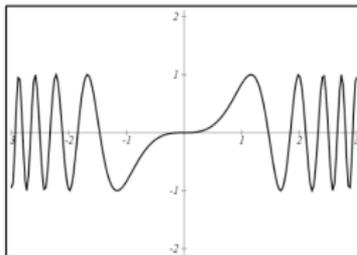
vysoká diskriminace

vysoká náhodnost

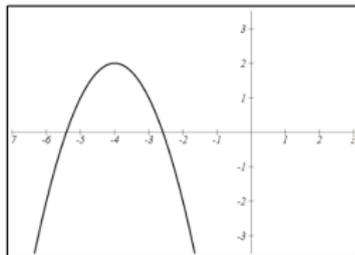
"na jistotu"



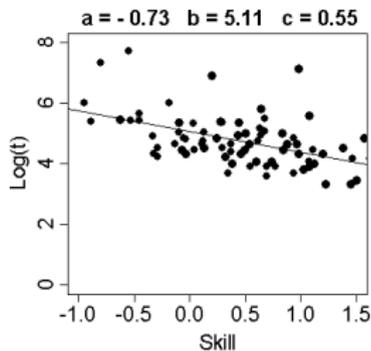
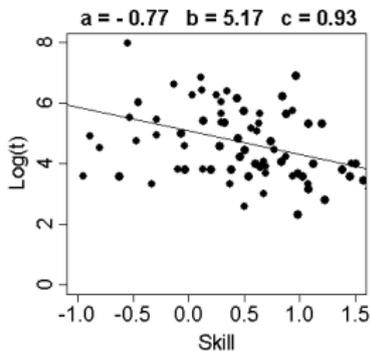
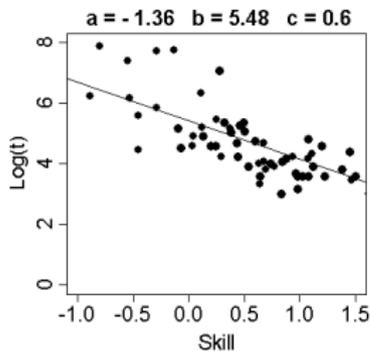
$\text{abs}(\log(\text{abs}(x)))$



$\sin(x^3)$



$-(x+4)^2+2$



# Extensions

- learning
- variability of student performance
- automatic detection of cheating

# Recommendations

- problems recommended based on the predictions
- ad-hoc scoring functions:
  - similar difficulty
  - not solved previously
- not evaluated yet

- `slepemapy.cz`
- adaptive practice of geography knowledge (facts)
- tries to estimate prior knowledge
- choice of places to practice  $\sim$  recommendation (forced)



Vyber na mapě stát

 Švédsko

 Nevím

 Pokračovat



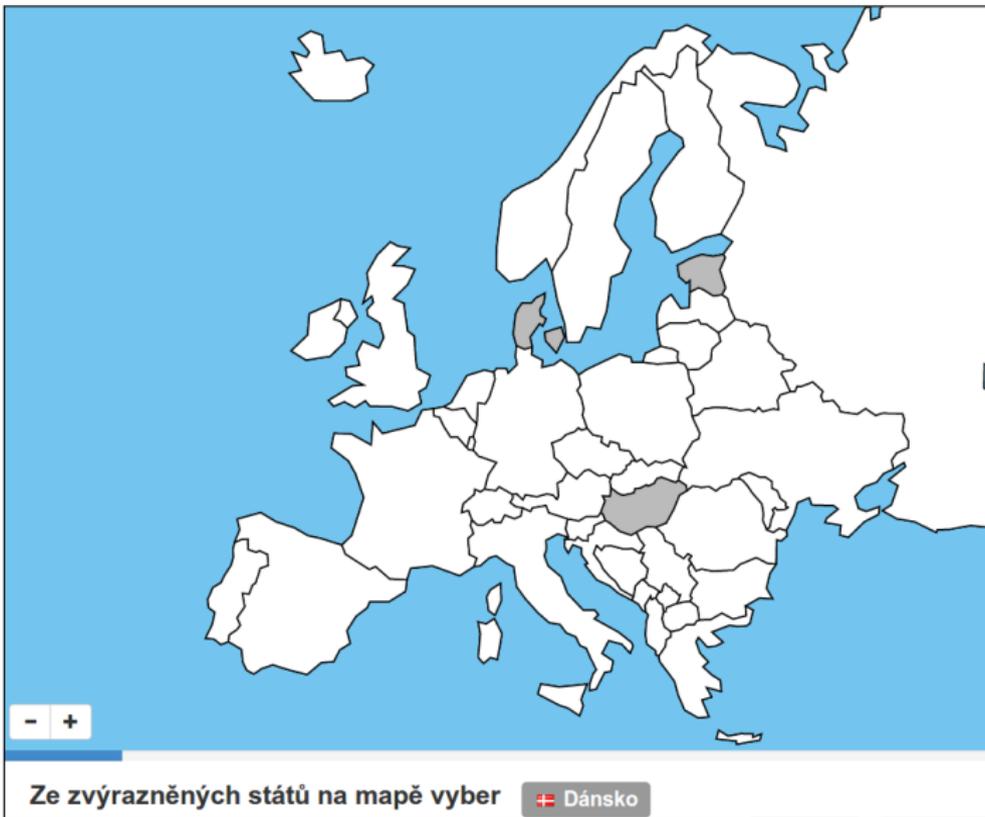
**Jak se jmenuje stát zvýrazněný na mapě?**

 Finsko

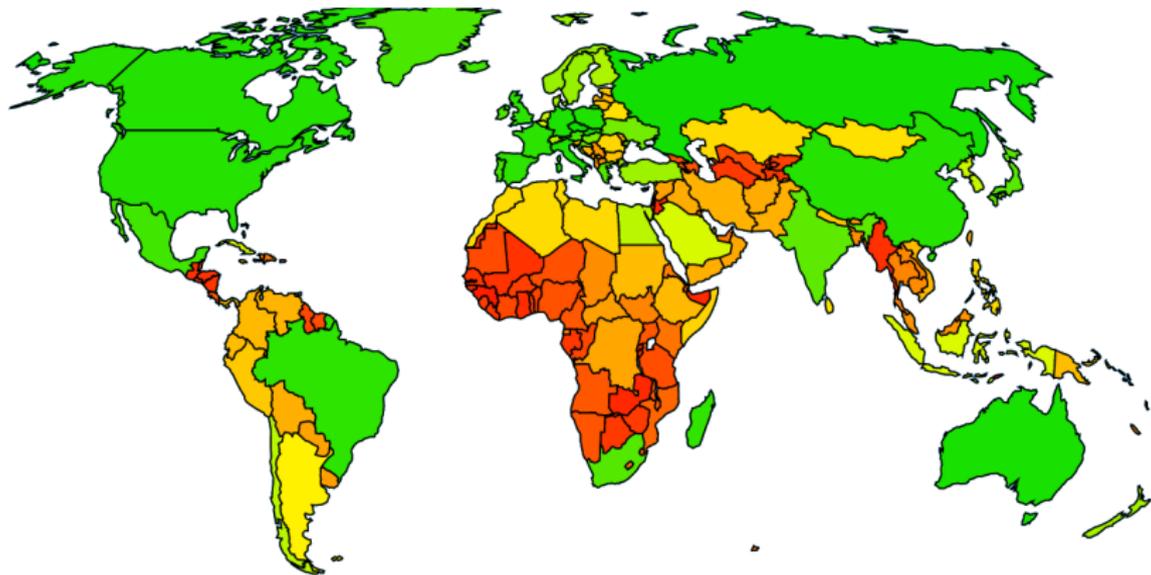
 Norsko

 Švédsko

 Nevím



# Geography – Difficulty of Countries



# Geography – Model

Model (prior knowledge):

- global skill of a student  $\theta_s$
- difficulty of a country  $d_c$

Logistic function:

$$P(\text{correct} | d_c, \theta_s) = \frac{1}{1 + e^{-(\theta_s - d_c)}}$$

# Geography – Model

- Elo system (originally from chess)

$$\theta := \theta + K(R - P(R = 1))$$

- magnitude of update  $\sim$  how surprising the result was
- related to stochastic gradient descent, “SVD” algorithm in collaborative filtering (but only single latent factor)

# Geography – Current Knowledge

- estimation of knowledge after sequence of answer for a particular place
- extension of the Elo system
- short term memory, forgetting implemented in simple way, work in progress

# Geography – Question Selection

question selection (based on predicted probability of correct answer)  $\sim$  item recommendation (based on predicted rating)

scoring function:

- predicted success rate, target success rate
- viewed recently
- how many times asked

# Geography – Multiple Choice Questions

- number of options – based on estimated knowledge
- choice of options – confused places

# Geography – Evaluation

- evaluation of predictions
  - offline experiment
  - issue with metrics: MAE, RMSE, AUC
  - data available for project
- evaluation of question selection (“recommendations”)
  - online experiment
  - issue with metrics: enjoyment vs learning

# Evaluation of Recommendations

experiments:

- comparison: our recommender algorithm vs random choice of questions
- comparison of different variants of the algorithm (different target success rate)

preliminary results: small differences, issues with data filtering

# Czech Grammar

- <http://www.umimecesky.cz/>
- adaptive practice for Czech grammar
- just starting, testing welcomed

**Umíme** česky

**Řešení úloh**

Přehled témat

Moje chyby

Žebříček

Které je správně?

paranoja

1

chromosom

2

potenciální

3

fotball

4

Umíme česky

Řešení úloh

Přehled témat

Moje chyby

Žebříček

Vstal pomalu neohrabaně s pocitem zoufalství.

Správně

1

Špatně

2

# Czech Grammar

TÉMA	SCHOPNOST	V ŘÁDĚ	VYŘEŠENO	
Mód mix všeho		42	439	<a href="#">Procvičit</a>
Čárky: souvětí		3	21	<a href="#">Procvičit</a>
Čárky: věta jednoduchá		7	27	<a href="#">Procvičit</a>
Délka samohlásek		5	30	<a href="#">Procvičit</a>
í/y koncovky podstatných jmen		6	32	<a href="#">Procvičit</a>
í/y koncovky přídavných jmen		6	21	<a href="#">Procvičit</a>
í/y koncovky sloves		8	32	<a href="#">Procvičit</a>

# Czech Grammar

## Moje chyby

TÉMA	SCHOPNOST	
<b>Čárky: souvětí (2)</b>		<a href="#">Procvíčit</a>
Přijedu, <u>a</u> nebo zavolám.	Přijedu, <u>a</u> nebo zavolám.	
Samozřejmě <del>(t)</del> že jsem byla zase nejlepší.	Samozřejmě že jsem byla zase nejlepší.	
<b>Čárky: věta jednoduchá (1)</b>		<a href="#">Procvíčit</a>
Jel na kole <del>(t)</del> ani ne hodinu.	Jel na kole ani ne hodinu.	
<b>Délka samohlásek (3)</b>		<a href="#">Procvíčit</a>
sc <del>(é)</del> nárísta	scenárísta	
fídicí pracovník	fídicí pracovník	
oxym <del>(e)</del> ón	oxymóron	

# Czech Grammar – Recommendations

- prediction of probability of success, variant of the Elo system
  - skill of students (per concept)
  - difficulty of questions
- recently viewed questions
- recommendation of concepts – least practiced, recent, ... (in progress)

# Wayang Outpost

- *A Multimedia Adaptive Tutoring System for Mathematics that Addresses Cognition, Metacognition and Affect*, 2014
- adaptive tutoring system for math
- Wayang Outpost → MathSpring,  
<http://mathspring.org/>
- specific feature: focus on affect and metacognition

# Wayang Outpost

The interface is titled "Expressions with Variables" and shows a "Skill Level" progress bar at 50%. The main text reads: "Dion wants to earn a minimum quiz average of 92% in his biology course. His grades so far are 89%, 95%, and 85%. Which inequality below represents the possible scores for his next quiz which will allow Dion to achieve his goal?"

The solution process is shown as follows:

$$\frac{\text{Sum of the values}}{\text{Number of values}} \geq 92 \longrightarrow \frac{89 + 95 + 85 + x}{4} \geq 92$$

Solve for x.

$$269 + x \geq 368$$
$$\cancel{269} + x - \cancel{269} \geq 368 - 269$$

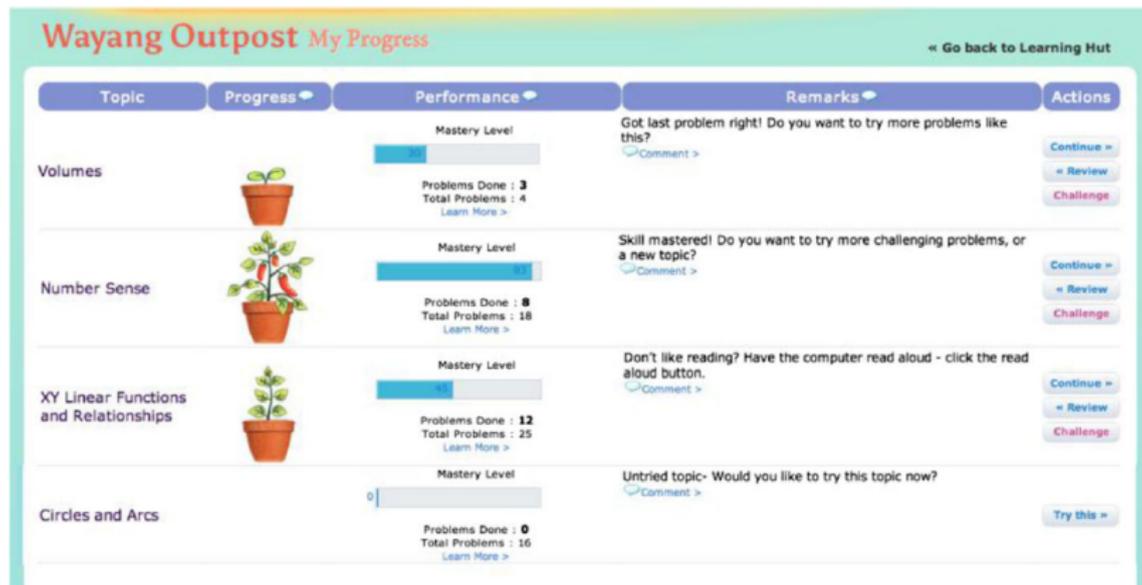
Four multiple-choice options are listed:

- (A)  $\{x \mid x > 99\}$
- (B)  $\{x \mid x < 99.5\}$
- (C)  $\{x \mid x \geq 99\}$
- (D)  $\{x \mid x \leq 99.5\}$

The interface includes a vertical toolbar on the left with icons for "HELP" and "DRAW", a "Skill Level" progress bar at the top, and a control panel at the bottom with buttons for "Formulas", "new problem", "resources", and "village". An animated character is seated at a desk on the right, with a control panel labeled "problem\_553" containing "Go To", "Hide me", and "Mute" buttons.

Fig. 1 The Wayang Outpost Math Tutor interface. An animated companion provides individualized comments and support

# Wayang Outpost: Open Learner Model



**Fig. 9** The open student model in Wayang is called the Student Progress Page (SPP). It encourages students to reflect on their progress for each topic (column 1). The plant (column 2) demonstrates the tutor's assessment of student effort, while the mastery bar (column 3) records presumed knowledge (according to Bayesian Knowledge Tracing). The tutor comments on its assessment of the student's behavior (column 4) and offers students the choice to continue, review or challenge themselves and make informed decisions about future choices (column 5)

# Wayang Outpost: Affect, Metacognition

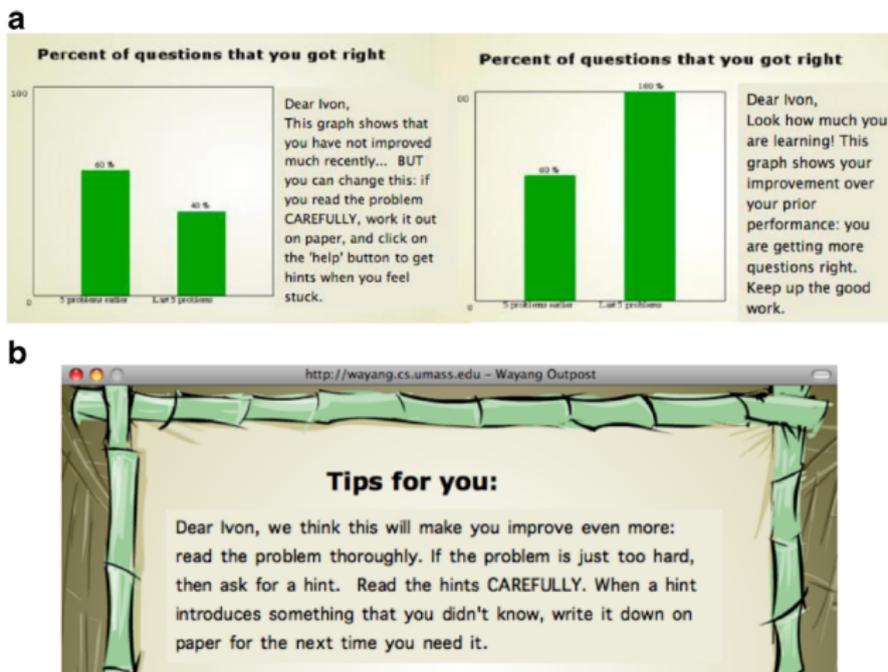


Fig. 11 a. Progress Charts in Wayang show students the accuracy of their answers. b. Tips in Wayang encourage good learning habits

# Wayang Outpost: Affective Learning Companions



**Fig. 14** Animated pedagogical agents display a range of emotions. Companions act out their emotion and resolve negative ones, expressing full sentences of affective and metacognitive nature, to support growth of mindset towards the view that intelligence is a state (and thus changeable)

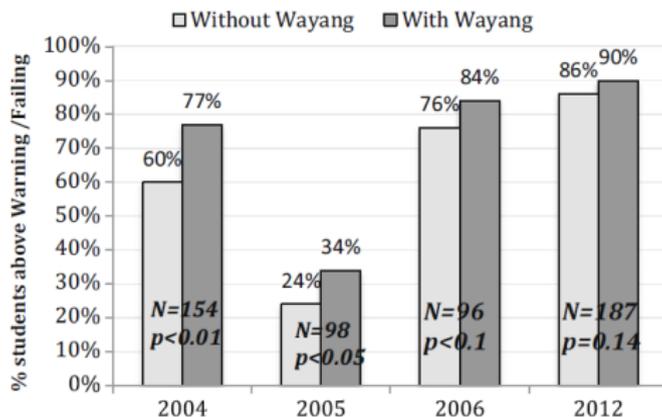
# Effort Based Tutoring

**Table 1** The effort-based tutoring algorithm informs pedagogical moves and affective decisions (last two columns) for each student on each problem. The algorithm first infers a reason for students behavior (fourth column) based on the number of incorrect student answers, hints requested and the amount of time spent (first three columns). Then the algorithm decides which pedagogical action the tutor should take (last two columns). The algorithm encourages transfer of student knowledge to subsequent questions of similar difficulty (rows 2, 4, 9), encouraging students to transfer skills and “fade” their need for help

Observed behavior and inferred reason for this behavior			Pedagogical Model Moves Cognitive or Affective or Metacognitive		
Incorrect	Hints	Time	Most Likely Reason	Decision	Affective/Metacog. Decisions
1 $< E (I) - \delta_{IL}$	$< E (H) - \delta_{HL}$	$< E (T) - \delta_{TL}$	Mastery without effort	Increase Problem Difficulty	Show learning progress
2 $< E (I) - \delta_{IL}$	$< E (H) - \delta_{HL}$	$> E (T) + \delta_{TH}$	Mastery with high effort	Maintain Problem Difficulty	Affective feedback: Praise Effort
3 $< E (I) - \delta_{IL}$	$> E (H) + \delta_{HH}$	$< E (T) - \delta_{TL}$	Hint abuse, low effort	Reduce Problem Difficulty	Deemphasize importance of immediate success
4 $< E (I) - \delta_{IL}$	$> E (H) + \delta_{HH}$	$> E (T) + \delta_{TH}$	Towards mastery, effort	Maintain Problem Difficulty	Praise effort
5 $> E (I) + \delta_{IH}$	$< E (H) - \delta_{HL}$	$< E (T) - \delta_{TL}$	Quick guessing, low effort	Reduce Problem Difficulty	Deemphasize importance of immediate success
6 $> E (I) + \delta_{IH}$	$< E (H) - \delta_{HL}$	$> E (T) + \delta_{TH}$	Hint avoidance and high effort	Reduce Problem Difficulty	Offer hints upon incorrect answer in the next problem
7 $> E (I) + \delta_{IH}$	$> E (H) + \delta_{HH}$	$< E (T) - \delta_{TL}$	Quick guess and hint abuse	Reduce Problem Difficulty	Deemphasize importance of immediate success
8 $> E (I) + \delta_{IH}$	$> E (H) + \delta_{HH}$	$> E (T) + \delta_{TH}$	Low mastery and High Effort	Reduce Problem Difficulty	Emphasize importance of effort and perseverance
9 Otherwise	Expected Behavior	Maintain Problem Difficulty			

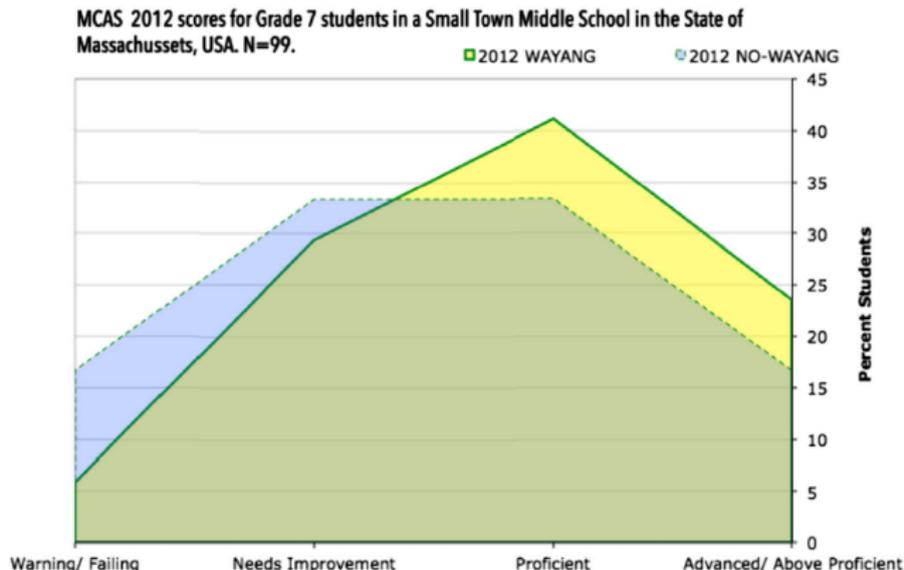
Note: Expected response (correct, hints, time) based on answers of other students  $\sim$  collaborative filtering

# Wayang Outpost: Evaluation



**Fig. 4** Massachusetts Statewide Standardized Test (MCAS) passing rates for experimental groups (using Wayang, dark grey) and control groups (in regular math class, light grey), within the same school, same grade and same teachers. Passing rates include several ratings above warning/failing

# Wayang Outpost: Evaluation



**Fig. 5** Area chart comparison of performance for a 7th grade of students on the Massachusetts Comprehensive Assessment System (MCAS), for students using vs. not using Wayang Outpost. Students represented by the *yellow/green polygon* used Wayang Outpost and students represented by the *blue polygon* did not use the tutor. Distribution of students using Wayang Outpost shifts to the right indicating that more students passed the exam and received a grade of “proficient” or “advanced” when using Wayang Outpost. Groups of students were matched in terms of teacher of seventh grade students

# Wayang Outpost: Evaluation

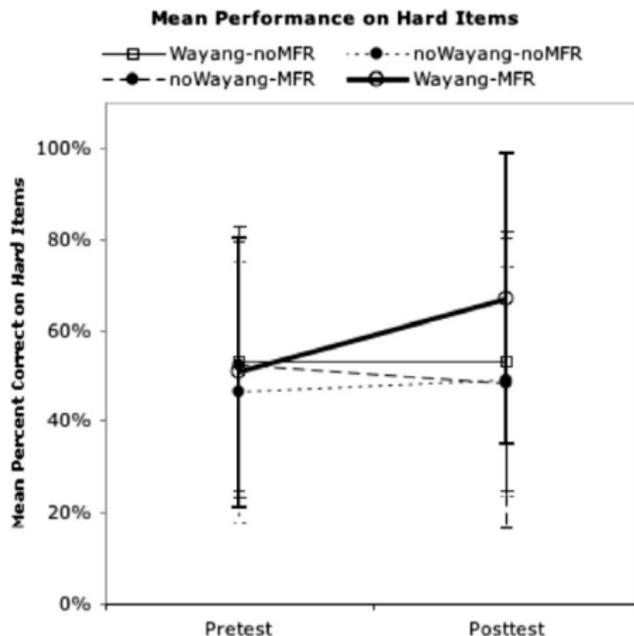
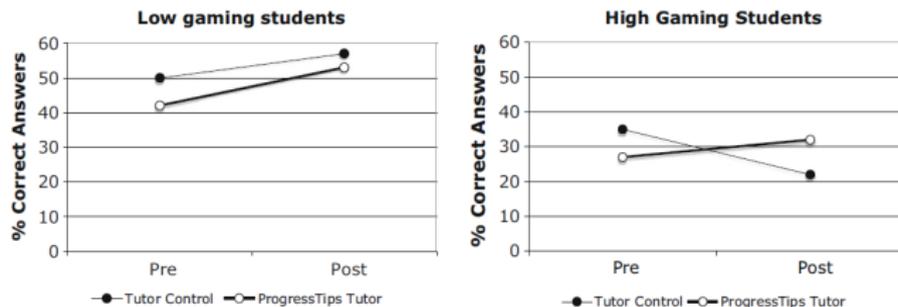


Fig. 7 Mean improvement (and standard deviations) on hardest items of the math pre/posttest. The *thick line* represents students who received both the Wayang Tutor and math facts retrieval training software; all other groups did not really improve on these harder multi-step items

# Wayang Outpost: Evaluation

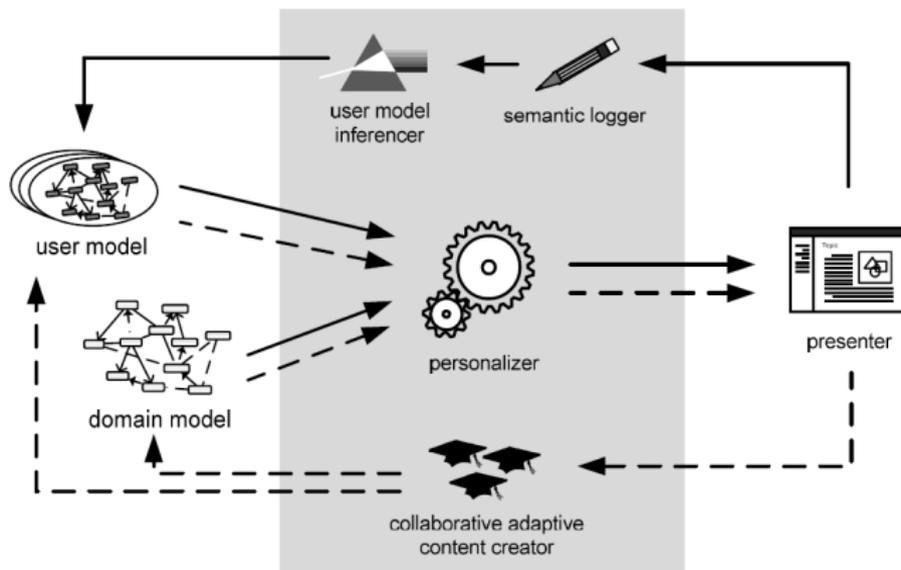
**Table 2** Students in the experimental group (last row) received tips and charts every 6 problems. Means and standard deviations in performance measures before and after tutoring for the three groups

Group	Math Pretest	Math Posttest	Passing Rate in State Standard Exam
No Tutor Control			76 % ( $N=38$ )
Tutor Control	40 % (20) ( $N=40$ )	40 % (28)* ( $N=40$ )	79 % ( $N=34$ )
ProgressTips Tutor	33 % (19) ( $N=36$ )	42 % (22)* ( $N=36$ )	92 % ( $N=24$ )



**Fig. 12** High gaming students improve math performance when they receive progress tips and interventions (*left*) but not when they don't receive interventions (*right*)

- PeWe (Personalized Web) Group at UISI FIIT STU, Bratislava
- adaptive education (mainly) for programming exercises



ALEF: A Framework for Adaptive Web-Based Learning 2.0, Šimko, Barla, Bielíková

**1** Odporúčaná

Funkcia FIRST  
Funkcie APPEND a LIST  
Špecifikácia typu zoznam  
Elementárne operácie

**2** Zvoľte si tému

- Paradigmy programovania
- Výrazy
  - Výrazy a príkazy
  - Vlastnosti špeciálnych výrazov
- Funkcionálne programovanie
  - Základné prvky jazyka lisp
  - Lisp-zoznam
  - Programovacie techniky
  - Pohľad na rekureziu

Chat (15)

Nezabudnite, že takmer každý výzbový text obsahuje niekoľko otázok, pomocou ktorých získate spätnú väzbu o vašich znalostiach.

## Funkcia REST

Komplementárnou funkciou k FIRST je funkcia REST, ktorá vráti celý zvyšok zoznamu bez prvého prvku. Poznamenajme, že funkcia REST vždy vráti zoznam.

Obr. 1 znázorňuje príklad použitia oboch funkcií FIRST aj REST.

```

  graph LR
    A["(7 2 14)"] --> B["FIRST → 7"]
    A --> C["REST → (2 14)"]
  
```

**5** Filtrik kapital: hodnotenie: +5

Aplikácia funkcie REST na prázdny zoznam je predtým definovaná v Common Lispe a vracia typ bodka-a-dvojica

Funkcie FIRST a REST. Aplikácia funkcie REST na prázdny zoznam a atóm nie je definovaná. Funkcie FIRST a REST môžeme kombinovať a tým vytvoriť ďalšie výberové operácie. Napr.:

```
(FIRST (REST '(7 2 14)))
2
```

**3** Príklad firstk

Zadanie:  
Definujte funkciu, ktorá vráti prvých K prvkov zoznamu.

(Element 2 '(a b c)) : -> (a b)  
(Element 0 '(a b c)) : -> NIL  
(Element 7 '(a b c)) : -> (a b c)

[Poznám odpoveď](#) [Nepoznám odpoveď](#)

◀ Prechádzajúci Nasledujúci ▶

**4** Otázky

Vyhodnot nasledujúcu formu  
(FIRST '(A B ( )))

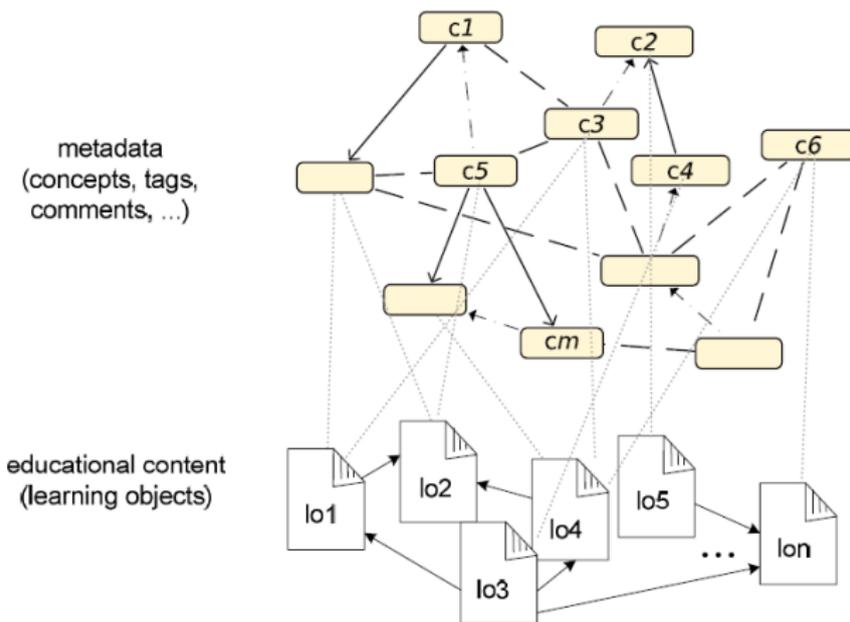
Odpoveď:

- (A) ✗
- NIL
- A ✓
- (A B)

Práča sa o otázku? Áno / Nie

Znam otázky Náhodná otázka

Otázky a odpovede



# CourseRank

- <https://www.courserank.com/>
- course evaluation and planning social system
- ranking of courses, grade distribution, other statistics
- recommendations
- originally Stanford, later many (US) universities
- similar features e.g. in Coursera

# Summary

personalized education  $\leftrightarrow$  recommender systems

- many similarities
- specific challenges
- difficult evaluation