

Concept Space Rating for Personalization of Learning Materials Based on Relations

Petr Šaloun, Zdenek Velart
Ostrava, Czech republic

Content

- Concept space / Learning object space
- Concept space evaluation
- Navigation
- Metrics
- System

Concept space

- concept represents a basic knowledge fragment
- relations between concepts (subclassOf, prerequisite, requires, ...)

Learning object space

- learning material, examples, multimedia, other learning content
- learning object has relation to one or several concepts

Basic ideas

- relations in concept space reflect relations in learning object space
- concepts has defined rating based on its position and relations in concepts space
- learning object has relation to one or more concepts
- test question has relation to one or more concepts
- learning objects can be ordered according their concept sets

Concept space evaluation

- Page rank algorithm
 - concept's rating based on ratings of related concepts and their number of relations
- Concepts space evaluation algorithm
 - based on page ranking algorithm
 - distinguish between relations types in the concept space
 - structural – subclassOf, partOf, is-a, ...
 - dependency – prerequisite, requires, ...

Concepts space rating algorithm

- relations evaluation $r_i.rating = \frac{count(r_i)}{N}$

- concept rating

$$c_i.rating = \frac{1-d}{N} + d \times \left(\sum_{c_j \in c_i.related} \frac{c_j.rating}{c_j.countr} + \sum_{c_k \in c_i.depend} c_k.rating \times r_k.rating \right)$$

Example

Learning object	Concept	Page rank value	Concept space rating algorithm value
Expression	atom	0.0026	0.0026
	number	0.0224	0.0311
	QUOTE	0.0036	0.0036
	s-expression	0.0072	0.0135
	NIL	0.0036	0.0036
	list	0.0111	0.0061
	ListFunciton	0.0100	0.0167
	T	0.0036	0.0036
Tail recursion	TailRecursion	0.0151	2.4263
Recursion	Recursion	0.0744	1.1436
Multiple recursion	MultipleRecursion	0.0139	2.6352

difference in values between concepts is due to dependency relations

Navigation example

Navigation menu structure:

- Leave course
- Navigation i
 - Výrazy
 - Seznamy
 - Přřazení
 - Predikáty a větvení
 - Rekurze
- Course menu i
 - Úvod
 - Základní typy
 - Seznamy
 - Výrazy
 - Predikáty a větvení
 - Přřazení
 - Rekurze
 - Rekurze na konci
 - Rekurze s jednoduchým testem ukončení
 - Rekurze s rozšířením
 - Rekurze s vícenásobným testem ukončení
 - Rekurze vícenásob
 - Závěreč

Základní typy jazyka LISP

Všeobecně se dá říct, že typy, které se používají v programovacích jazycích se dají rozdělit na *atomické* a *složené*. Mezi atomické zařazujeme čísla, boolovské hodnoty a řetězce.

Atomické typy

V LISPu rozdělujeme atomické typy na *čísla* (numerické atomy) a *symbols* (nenumrické atomy).

Čísla

na základě své reprezentace a implementace se dělí na *celá* a *reálná* čísla.

Symbols

se zapisují jako posloupnost písmen, číslic a speciálních znaků: +, -, *, /, _, ~, @, \$, %, ^, &, \, <, >. Příkladem symbolů může být VLAK. LISP nerozlišuje velká a malá písmena, proto symboly VLAK a vlak a VLAK se považují za stejné.

Boolovské hodnoty představují speciální symboly. Hodnota nepravda (false) odpovídá symbol NIL a hodnotě pravda (true) symbol T.

Složené typy

Složený typ údaje, který se používá se nazývá *seznam*. Seznamem nazýváme konečnou posloupnost prvků určitého typu. Seznam se zapisuje jako posloupnost prvků, uzavřených v kulatých závorkách a oddělených mezerou. Prvkem seznamu může být libovolný s-expr, tím pádem i jiný seznam. Příklad jednoduchého seznamu

```
(0 1 2 3 4 5 6 7 8 9)
(1 JEDNA 2 DVE 3 TRI)
```

Složitější seznam

```
(0 1 ((2 3) (4 (5 6) 7)) 8 9)
```

Prázdný seznam se označuje symbolem NIL (zároveň symbol - boolovská hodnota nepravda a prázdný seznam) nebo ().



I would like to try a test!



I understand it all
I'll click on the next page from menu.



I understand it all,
take me to the next part.

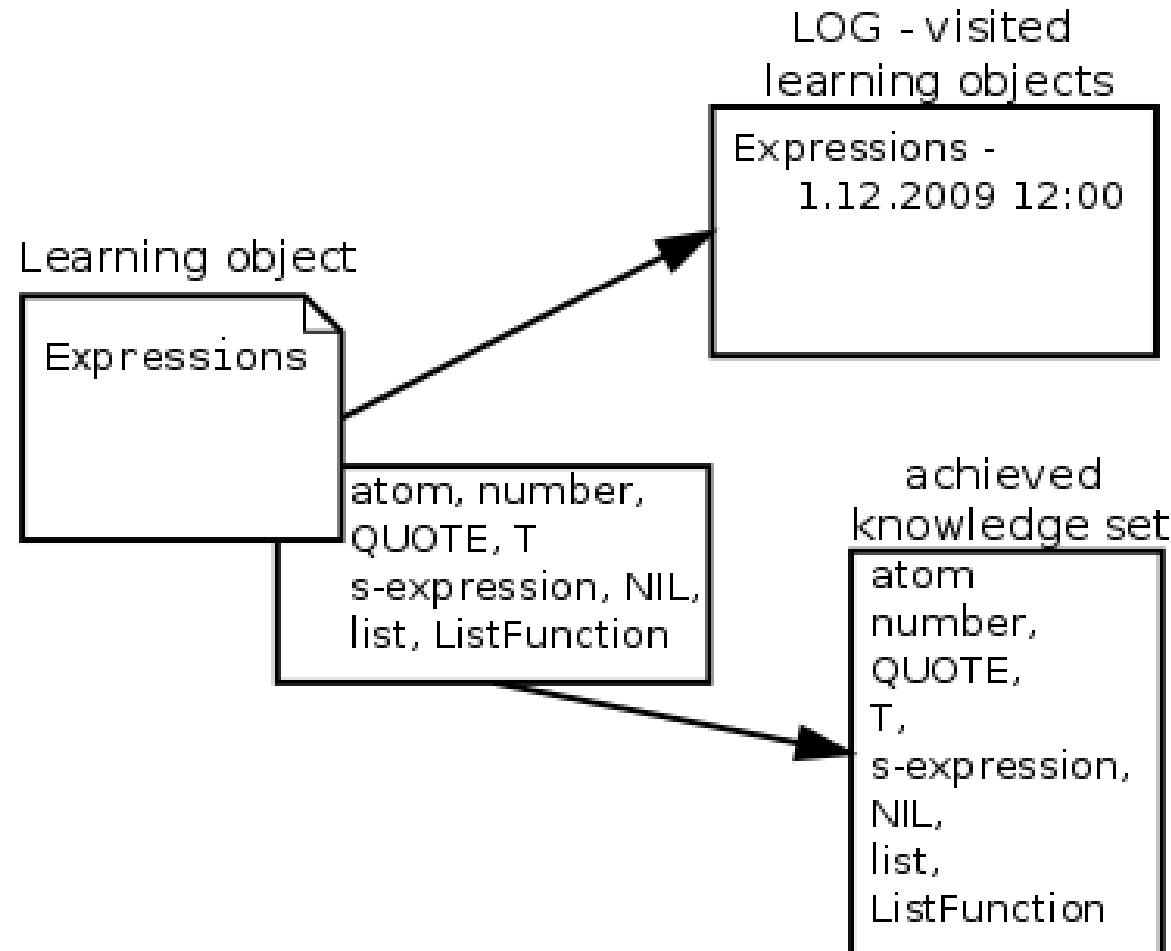
test from current learning material

“confirmation” of knowledge, student chooses next page from menu

“confirmation” of knowledge, student is taken to next page (first in menu)

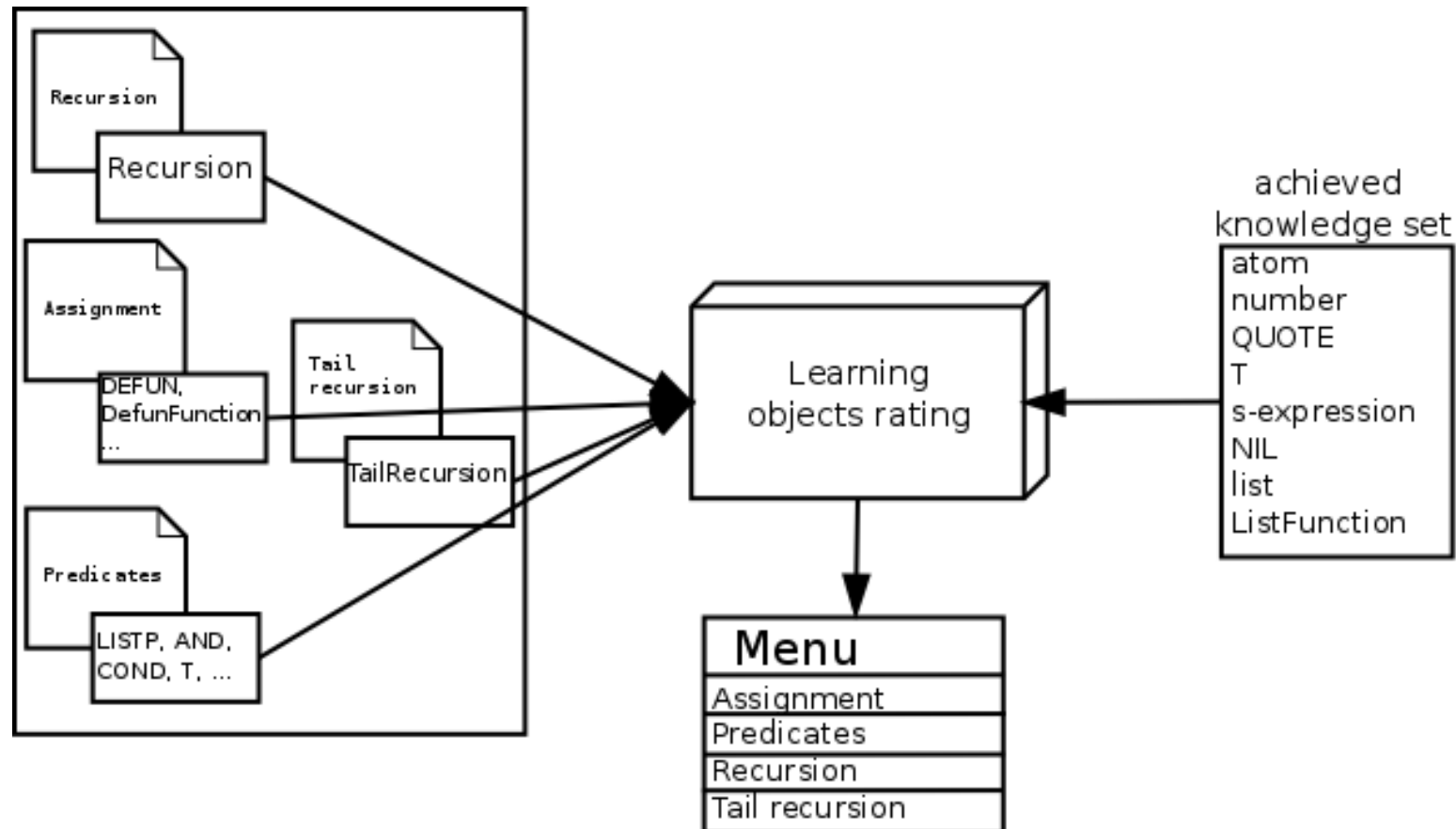
Navigation – user profile

- update “achieved knowledge” set
 - concepts from learning object concept's set are added
- Logging
 - name of the learning object together with timestamps is stored



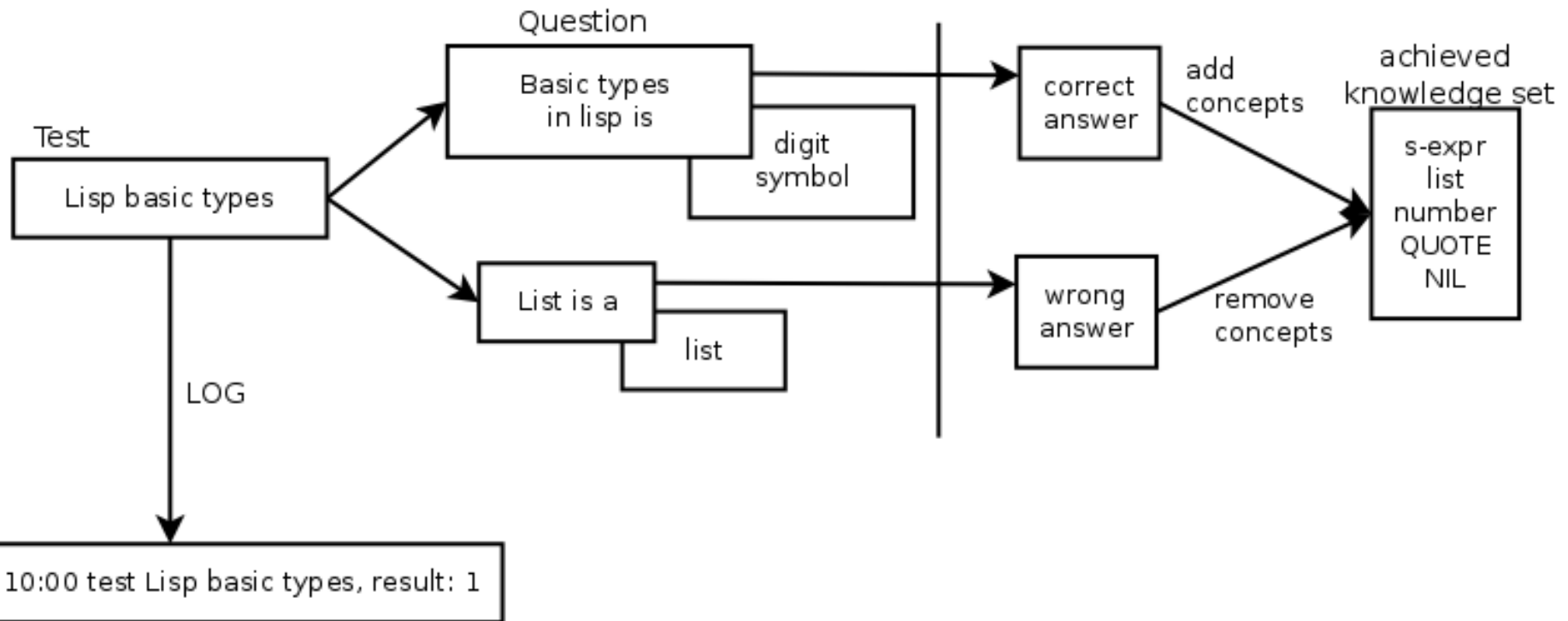
Navigation – menu creation

Menu creation according the “achieved knowledge” set and chosen metric



Navigation – test

- learning object test – question has a concept set
- correctness of answer to question determines if concepts are added to or removed from achieved knowledge set



Metrics

- order metric
 - learning object have predefined order, items in the menu are shown according this
- concepts set size
 - learning objects are ordered according size of it's concepts set
- unvisited learning objects rating
 - only unvisited learning objects
 - learning object rating is computed as sum from complement of the concept's set and achieved knowledge set
- all learning objects rating
 - similar as previous
 - to visited learning object rating a penalization is added, either constant or time varying
- similarity metrics
 - learning objects which concepts set has the biggest intersection with currently visited learning object concepts set are found

Example cont.

- navigation for learning object from previous example
- metric – unvisited learning objects rating
- student's achieved knowledge set is empty
- learning object set contains four learning objects – Expression, Tail recursion, Recursion, Multiple recursion

Page ranking	value
Multiple recursion	0.0139
Tail recursion	0.0151
Expression	0.0641
Recursion	0.0744

Concept space rating	value
Expression	0.0808
Recursion	1.1436
Tail recursion	2.4263
Multiple recursion	2.6352

System

- based on AHAM model
- user's model – achieved knowledge set, visited learning objects set, test and system log
- domain model – concept set with ratings, learning object set together with learning content (learning text, examples, multimedia, ...), tests
- adaptation model – algorithm for concept rating, metrics, test evaluation

Case study

- Lisp (12 learning object / 106 concepts)
- several groups of students
 - different concept evaluation (page rank, concept space rating algorithm)
 - different metrics
- students will take Lisp course
 - relatively unknown learning material to students – same starting conditions for everyone
 - final test for verification of learned knowledge
 - a questionnaire concerning the navigation
- university and high school students
- evaluation is running now, we expect results as a Christmas gift

Contribution

- concept space rating algorithm
- navigation
- metrics

Future work

- offering more examples to student after unsuccessful test, extending the learning material
- factor of forgetting
 - rating of visited learning object is time-dependent
 - value will increase with the time

Thanks for your
attention!

Questions?

University of Ostrava, VSB – Technical University
petr.saloun@osu.cz, zdenek.velart@gmail.com