

LKB-FOS Update

John Carroll

Department of Informatics, University of Sussex, UK

DELPH-IN Summit, July 2020

Outline

Functionality: [incr tsdb()], maxent models, LSP

User interface

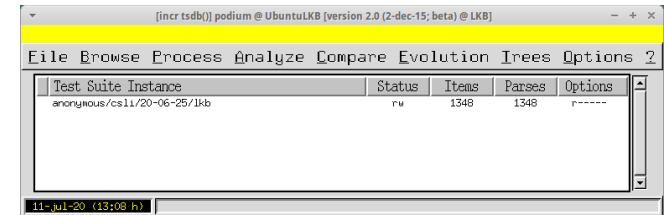
Scalability: experiments

Internals: memory leaks, bugs fixed, distribution

Functionality

[incr tsdb()]

- Most of [incr tsdb()] ported, including the 'podium'
- At present, only for Linux; for macOS, would need to recompile the C code of tsdb and swish++
- No LOGON-specific functionality available (i.e. source code enabled by the :logon feature)
 - no PVM, WWW demo, SVMs, language models, or external MT system interfaces
- Start the podium with (tsdb:tsdb :podium)



Maxent models

- Create maximum entropy disambiguation models by calling `tsdb::train` or with the `[incr tsdb()]` podium command `Trees -> Train`
- For training, need `tadm` and evaluate binaries from LOGON distribution
- Selective unpacking integrated into the parser and generator; to load and use maxent models to rank results:

```
#+:tsdb
(tsdb::read-model (lkb-pathname (parent-directory) "redwoods.mem"))
#+:tsdb
(setf *unpacking-scoring-hook* #'tsdb::mem-score-configuration)
```

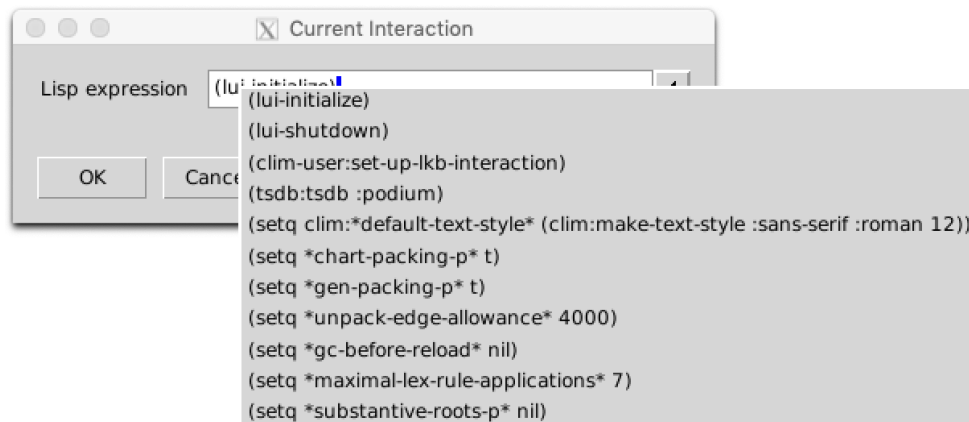
- Only local trees (mother and daughters) scored; no 'grandparent' features yet

LSP

- Made the Linguistic Server Protocol fully functional, allowing LKB-FOS to be run as a server over a socket connection

User Interface

- Rewrote double-clickable macOS LKB.app: better error checking, avoids dependencies on Finder (c.f. macOS Mojave 10.14 onwards)
- In the LKB Top Advanced menu, added 'Evaluate lisp expression...'; the associated dialog is preloaded with useful commands:



- Enabled the Edit command in the Transfer Output window Debug menu; allows an MRS to be edited in emacs
- Many minor interface fixes

Scalability

- Changed some data structures for faster update/lookup on large inputs, and/or better memory locality:
 - passive edge charts in both parser and generator
 - selective unpacking edge-local agendas and decomposition indices sets
 - chart edge ‘registry’
- Changed parser agenda edge priority function to make packing more compact (also probably improves memory locality)
 - separate priorities for exhaustive full-forest and non-exhaustive modes
- Better handling of edge, agenda and sentence length resource limits

Parsing: Rondane (1424 items), ERG 2018, compute top-ranked parse

Start-symbols `root_strict`, `root_informal`, `root_frag`, `root_inffrag`, `root_robust` (only the first 4 of these in PET), chart mapping (not LKB-FOS), resource limits giving ~25 failures, packing, selective unpacking with `redwoods.mem`. Total numbers of parses are comparable.

	<i>CPU (mm:ss)</i>	<i>Memory (GB)</i>
ACE *	21:29	176
ACE (with <i>--disable-generalization</i>) **	34:15	252
LKB-FOS ***	51:43	398
PET ****	120:11	~270
<i>agree</i> *****	75:22	

agree test on a 3.2GHz 16GB machine; all other tests on an iMac 3.3GHz i5, 16GB: macOS 10.14, and Ubuntu 3.19.0-79-generic in VirtualBox

Generating: Rondane (first 200 MRSes), ERG 2018, compute top-ranked string
Start-symbol `root_gen`, same resource limits, packing, selective unpacking with `redwoods.mem`

	<i>CPU (mm:ss)</i>	<i>Memory (GB)</i>
ACE [†]	30:57	45
LKB-FOS [‡]	5:35	47

* version 0.9.30, macOS, successfully parsed 1353 items, options `-1 -R`

** successfully parsed 1354 items, options `-1 -R --disable-generalization --max-chart-megabytes=2000 --max-unpack-megabytes=2500`

*** release 11-jul-2020, macOS, successfully parsed 1150 items, parameters `*first-only-p* 1, *maximum-number-of-tasks* 200000, *maximum-number-of-edges* 225000, *unpack-edge-allowance* 50000`

**** cheap 0.99.14svn_cm, Linux, successfully parsed 1215 items, rejected 7 items before parsing, options `--test -packing=15 -nsolutions=1 -cm -default-les=all -timeout=120`

***** with ERG 1214, Windows, attempting 1381 items (43 items removed due to hitting memory limit, and not included in the CPU time)

[†] successfully generated 108 strings, 21 items hit memory limit, options `-e -1 -R`

[‡] successfully generated 141 strings, 3 items hit edge limit, parameters `*gen-first-only-p* 1, *maximum-number-of-edges* 225000, *unpack-edge-allowance* 50000`

Internals

Memory leaks fixed (sample)

- acl-compatible library: dealt poorly with unexpected thread termination
- Chart display: a global variable intended for debugging was not conditionalised out; also, CLIM somehow held on to edge structures
- Parser agenda: sometimes retained pending tasks after parser exited
- Interface to yzui: often holds on to data after all associated yzui windows have been closed – issue identified but not yet fixed

Bugs fixed

- In some cases the generator failed to filter properly on `*gen-start-symbol*` and `*additional-root-condition*`
- In trigger rule application, FOS (and Classic) LKB ‘lost’ lexical entries compared to LOGON LKB, due to a difference in `extract-pred-from-rel-fs`

Distribution

- Removed out-of-date versions of third party open source Common Lisp subsystems; the build script uses Quicklisp library manager to load them
- LKB binaries now stored compressed and automatically uncompress themselves when loading, reducing storage requirements by 80%

Summary

Further improvements to LKB-FOS over the past year

Highlights

- substantial new functionality
- many internal improvements

Next steps

- unified grammar configuration
- chart mapping
- selective unpacking grandparenting
- Windows version