Tuning a Grammar to Improve Parsing Efficiency

Dan Flickinger

DELPH-IN Summit 2023

Galicia, Spain

29 June 2023
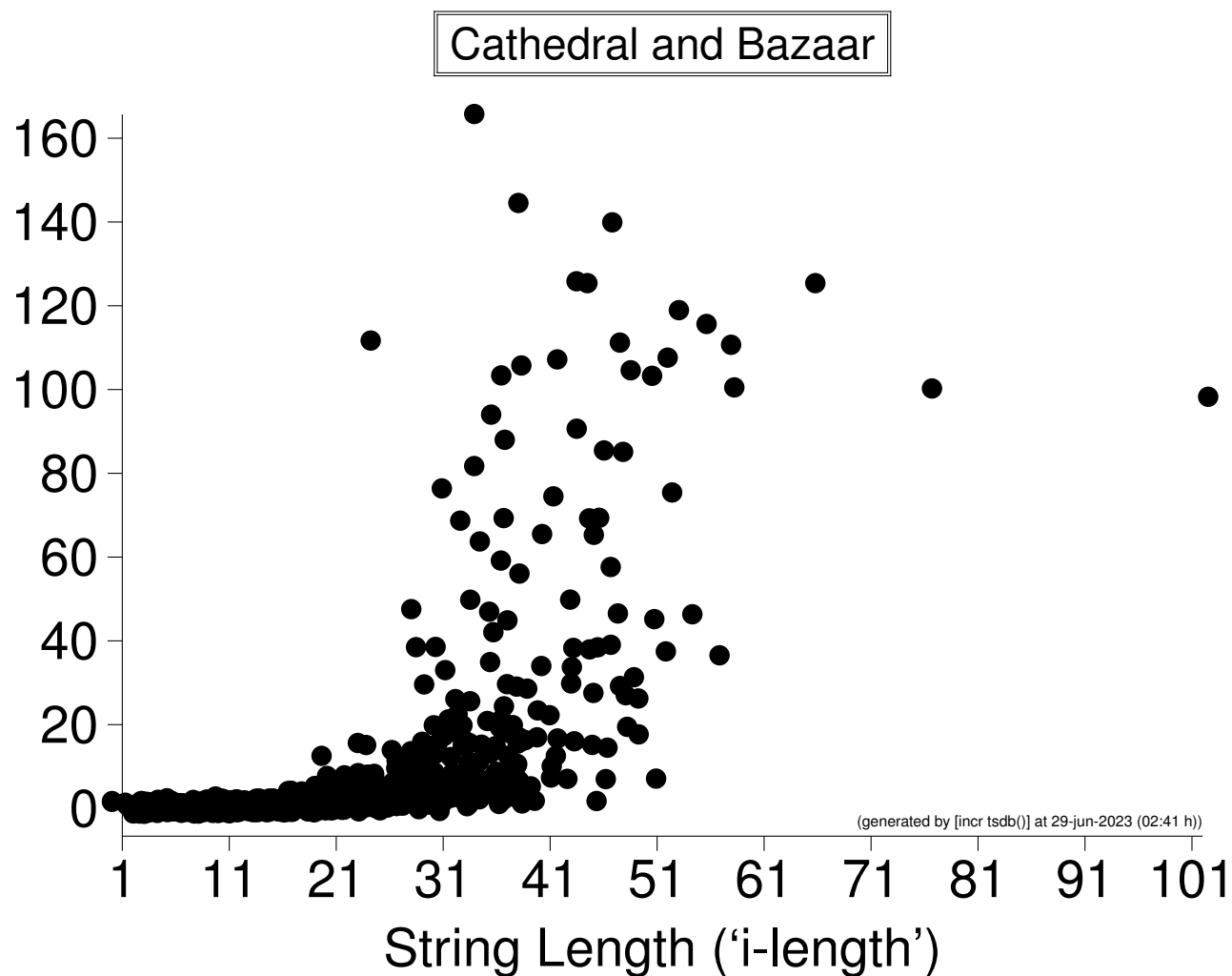
Methodology for improved efficiency

- Modify the grammar to expand coverage or fix overgeneration

- Reparse a treebanked corpus to confirm effects of changes

- Measure performance over the corpus

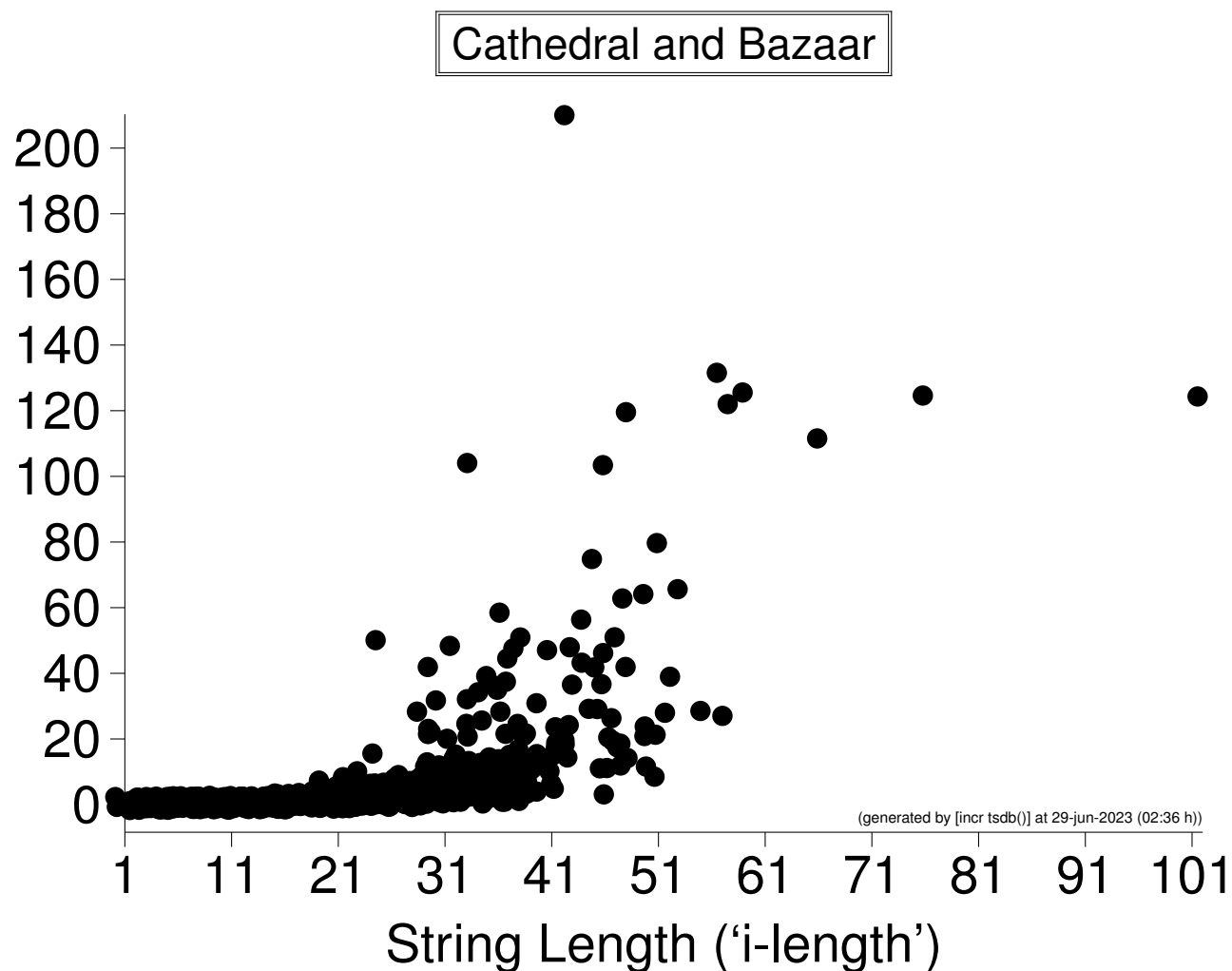# Methodology for improved efficiency

- Modify the grammar to expand coverage or fix overgeneration

- Reparse a treebanked corpus to confirm effects of changes

- Measure performance over the corpus

- Inspect/compare performance on a specific profile

- Identify specific sentence with surprising costs

- Hypothesize source of trouble, and adjust the grammar

- Confirm improved performance on that sentence

- Reconstruct treebanked sentences for that profile or full corpus

Measuring performance: [incr (tsdb)] on ERG 2022

Cathedral and Bazaar

String Length ('i-length')

(generated by [incr tsdb()] at 29-jun-2023 (02:41 h))

Measuring performance: [incr (tsdb)] on ERG 2023

Cathedral and Bazaar

(generated by [incr tsdb()] at 29-jun-2023 (02:36 h))

String Length ('i-length')

Examples of grammar tuning

- vmod rule: nonhd dtr PUNCT comma_or_rbc_or_pair_or_no_punct (exclude clause)

- head_second_comp rule: ditto

- hd-aj_int-sl_c: gap on nonhd dtr: HEAD basic_noun (exclude PP comps of nouns)

- if and other subord preps: no gap in complement

- cl-adv: only modify V and Adj – block for ”[one][thing we are concerned about]”

- block sbrd-prd from modifying if-clause and similar

- extracted_comp: add [RSUBJHD supnoun] to mother to block ”beside Kim are”

- split subconj ”to” and ”in order to” so POSTHD - modifies S, + modifies VP to avoid spurious ”we arise to arrive”

Available utilities

- Use [incr (tsdb)] to inspect/compare performance on profiles

- LUI to display a parse chart for a sentence

- `recons` script to reconstruct derivations for a profile

    `./recons -g erg.dat -v erg/tsdb/gold/hike;`

    Reports failure source for each item that fails to reparse

    Takes a couple of minutes for all of Redwoods (1.5M words)

- LKB-FOS's new parse chart summary and subsumption check