A Parent's Guide to Raising Grammars: Minding the Generation Gap

Michael Wayne Goodman
goodmami@u.washington.edu
with
Francis Bond
bond@ieee.org

University of Washington
and
NICT

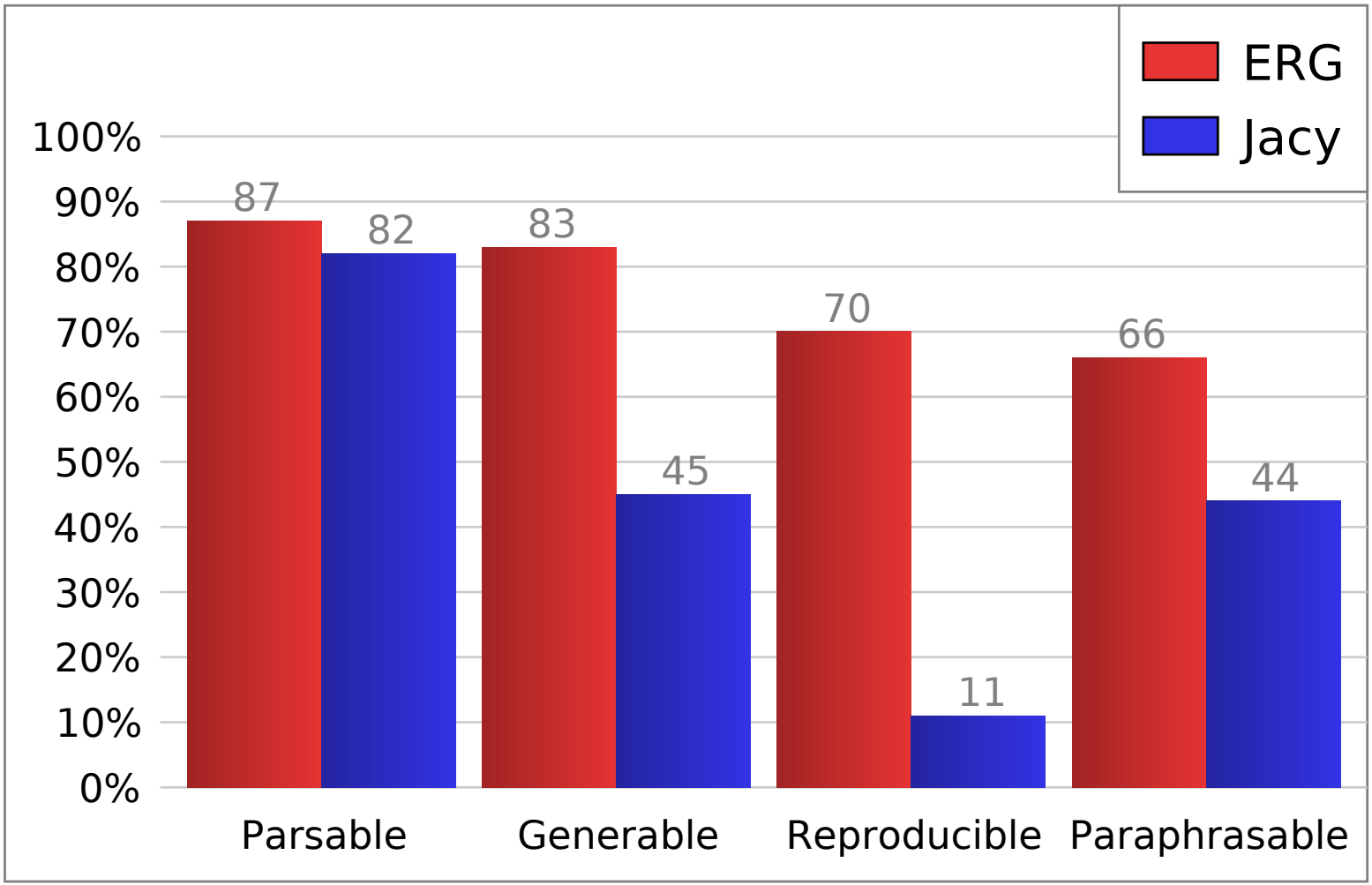
July 21, 2009

Motivation

- Paraphrasing (Bond et al., 2008)
- X-Ja Translation
- A more well-rounded, correct grammar (Flickinger, 2008)

Initial Results

Sibling Rivalry



We want our grammars to be:

- Well-behaved
- Successful
- Mature

Error Mining

Finding Faults, Just Like Mom

- Correcting errors is (still) the work of the grammar developer
- Finding errors doesn't have to be

Parse Errors

Immature Behavior

- Missing lexical items
- Incomplete grammar rules
- Bugs

Previous Work

- van Noord (2004) found common N-grams in unparsable sentences
- Sagot and de La Clergerie (2006) fine tuned the selection algorithm
- de Kok et al. (2009) improved support for longer N-grams

Generation Errors

Bad Behavior

- Does not generate
- Ungrammatical or incorrect generations
- Suboptimal ranking[1]

[1] Not necessarily a problem

Error Detection

General idea:

- Consider top parse as correct
- Look for differences in, or lack of, generations
- Asymmetries represent problems ("Generation Gap")
- Generation vs negative examples

Methodology

Information used:

- Top parse
- Top N generations

Information analyzed:

- Surface string
- Derivation tree
- MRS

Characteristics

Item characteristics:

- Parsable
- Generable
- Reproducible
- Paraphrasable
- (Utool validation)

Item Characteristics

Unparsable

“この 薬 は 筋肉 痛 を 和らげる”
“This medicine relieves muscle pain.”

Ungenerable

“花瓶 で 持ち なさい”
“Hold (it) in a vase.”

Not reproducible

“どうして 逃げ た の”
“Why did (you) run away?”

Characteristics

Comparative characteristics:

- Lexemes
- Rules
- Derivation Tree[2]
- Surface form[3]
- MRS

[2] Without root nodes

[3] Without punctuation

Comparative Characteristics:

Different lexemes from source

“彼 は 身分 の 高い 人 だ”
“そいつ は 身分 の 高い 人 だ”
“He has high social standing.”

Different derivation tree from source

“彼女 は 色 が 黒い”
“色 が 彼女 は 黒い”
“Her color is black.”

Different set of rules from source

“昨日 は とても 寒かっ た”
“* 昨日 は とても 寒く た”
“Yesterday was very cold.”

Comparative Characteristics (continued):

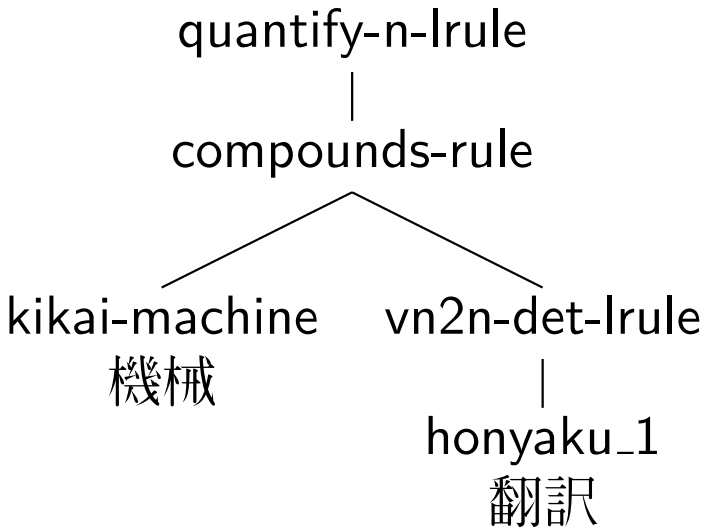
Different surface form from source

(any of the previous)

Different MRS from source

“あの　店　は　サービス　が　良い”
“あの　店　＝　サービス　が　良い”
“That store has good service.”

Extracting Rule Paths



- quantify-n-lrule → compounds-rule → kikai-machine
- compounds-rule → kikai-machine → 機械
- quantify-n-lrule → compounds-rule → vn2n-det-lrule
- compounds-rule → vn2n-det-lrule → honyaku_1
- vn2n-det-lrule → honyaku_1 → 翻訳

Derivation tree for “機械翻訳” (“machine translation”) and paths with length of 3

Building a Model

- MaxEnt-based classifier
- N-grams of derivation paths as features
- String of item characteristics as labels
- Find most salient feature (path) for a given label

Top 10 Cannot Generate

Score	Count	Rule Paths
1.4234	109	hf-complement-rule → quantify-n-lrule → compounds-rule
0.9601	54	hf-complement-rule → quantify-n-lrule → nominal-numcl-rule → head-specifier-rule
0.7562	63	head-specifier-rule → hf-complement-rule → no-nspec → "の"
0.7397	62	hf-complement-rule → head-specifier-rule → hf-complement-rule → no-nspec
0.7391	22	hf-complement-rule → hf-adj-i-rule → quantify-n-lrule → compounds-rule
0.6942	36	hf-complement-rule → hf-complement-rule → to-comp-quotarg → "と"
0.6762	82	vstem-vend-rule → te-adjunct → "て"
0.6176	26	hf-complement-rule → hf-complement-rule → to-comp-varg → "と"
0.5923	36	hf-adj-i-rule → hf-complement-rule → quantify-n-lrule → nominal-numcl-rule
0.5648	62	quantify-n-lrule → compounds-rule → vn2n-det-lrule

First fix: Topic Hierarchy

Topic Marker Problem

“あの　店　は　サービス　が　良い”
“あの　店　＝　サービス　が　良い”
“That store has good service.”

Second fix: Compounds

Compounding Problem

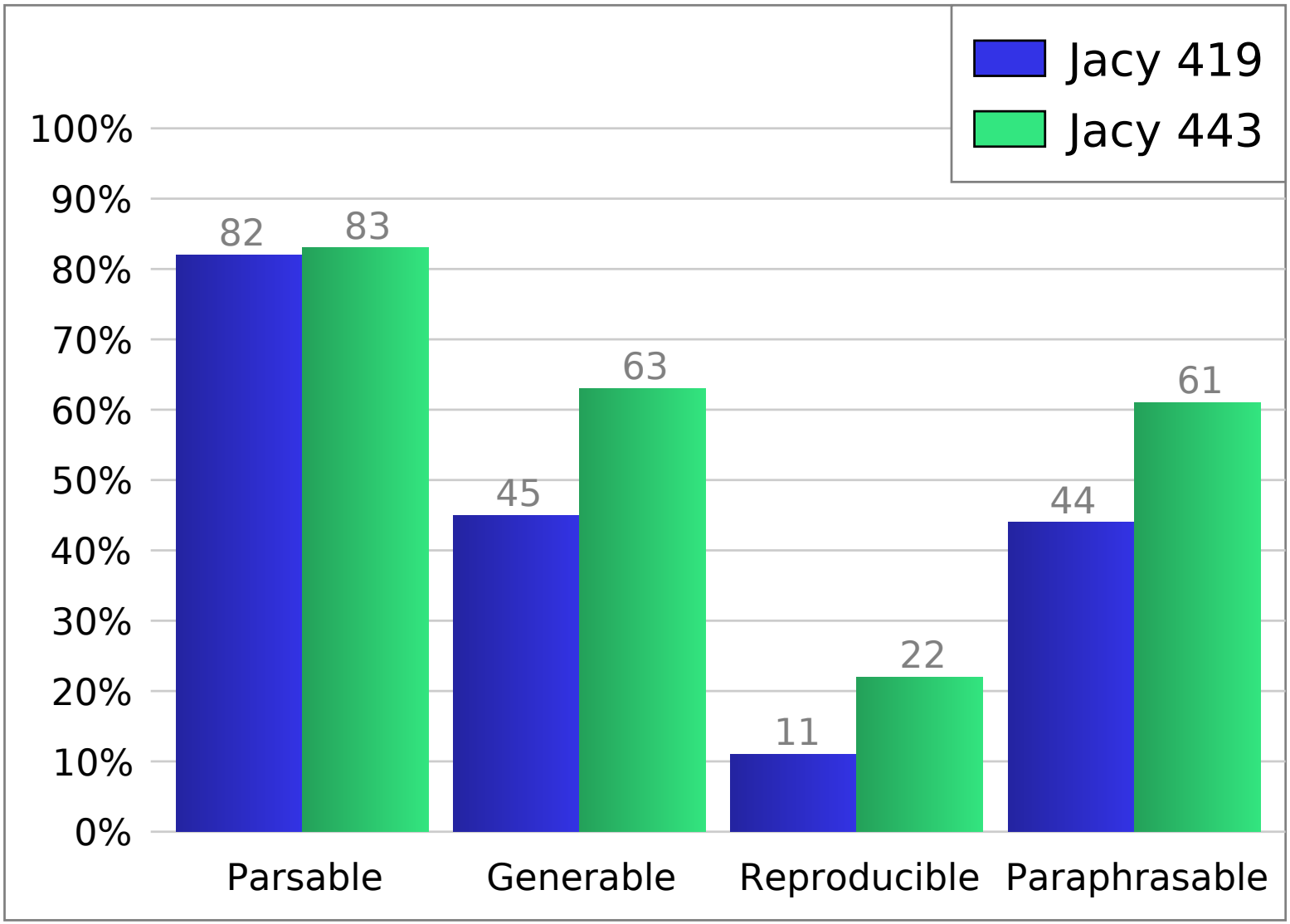
“彼　は　都会　生活　に　あこがれ　て　いる”
(Did not generate)
“He longs for the city life.”

Third fix: Numeral Classifiers

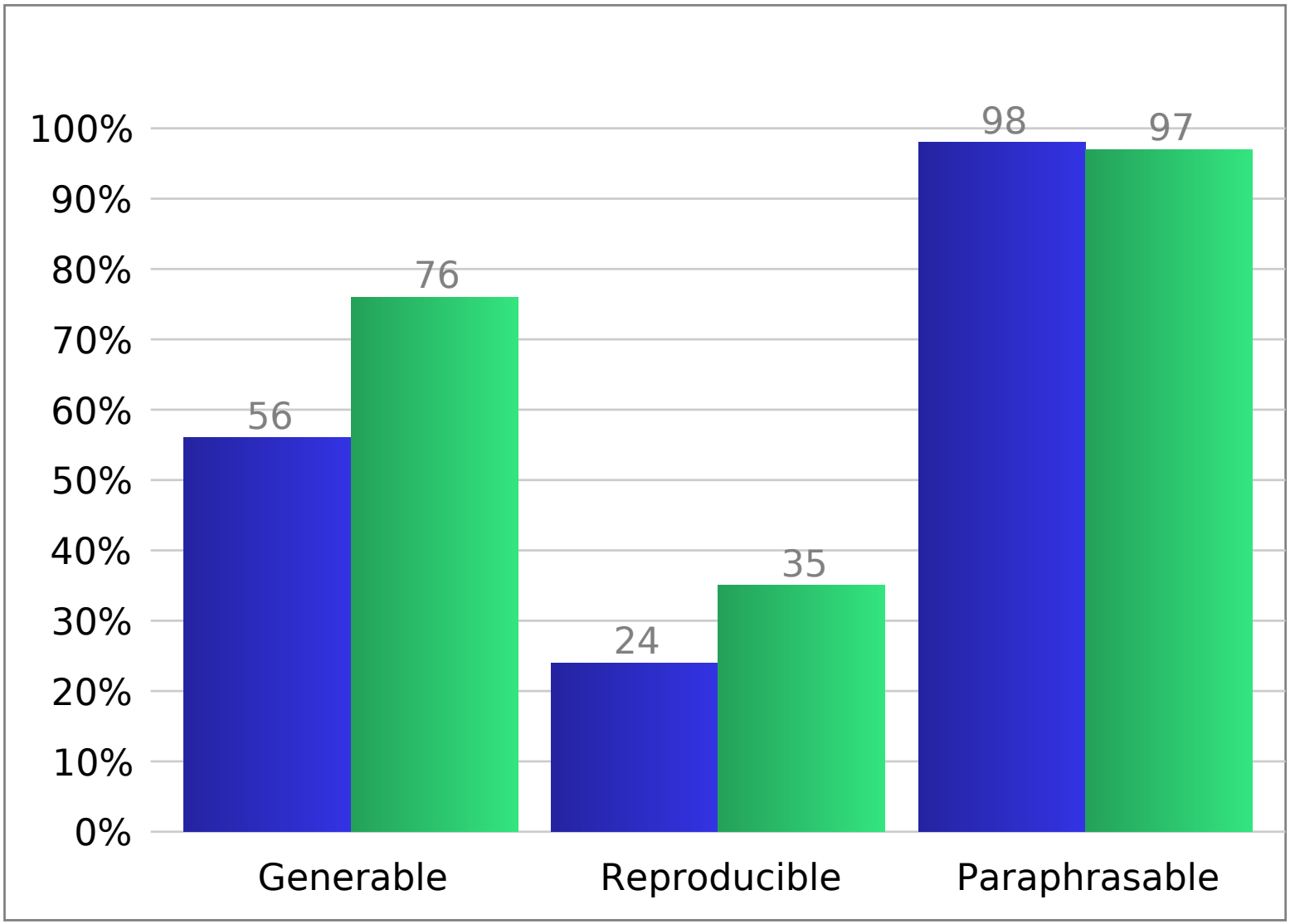
Classifier Problem

“私 の クラス に は 40 人 の 生徒 が い ます”
(Did not generate correctly)
“There are 40 students in my class.”

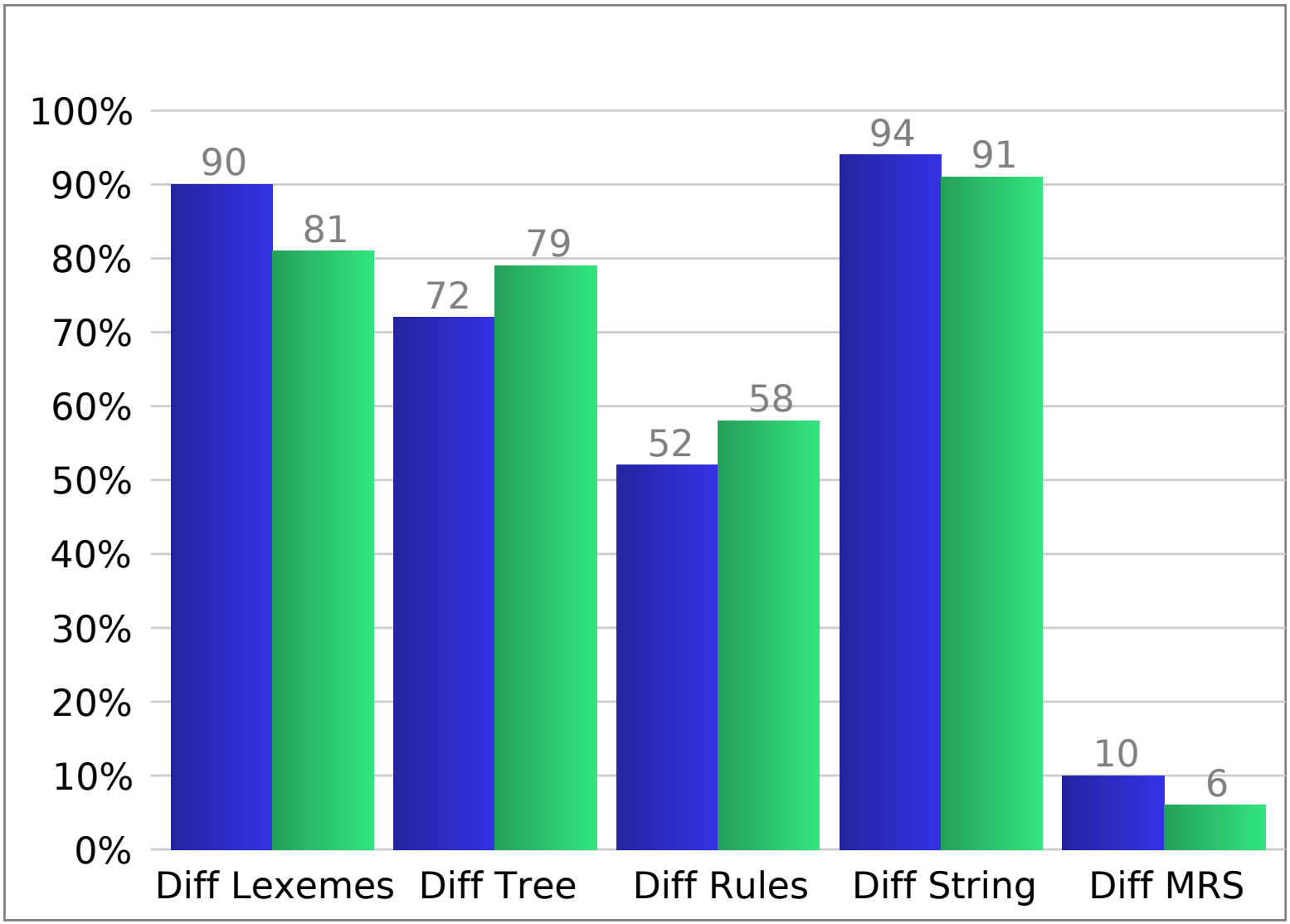
New Statistics: Jacy 419 – 441, absolute



New Statistics: Jacy 419 – 441, relative



New Statistics: Comparative characteristics



Conclusion

What we have:

- Successfully characterizes parsed and generated sentences
- Useful for finding grammar errors and inadequacies, paraphrases, and corpus errors
- Successfully determines rules responsible for certain characteristics
- Should work for most[4] HPSG implemented grammars

[4] Tested on Jacy, ERG, GG, and Wambaya

Conclusion

English examples with the ERG

Does not parse

It proved to be the deathblow to their plan.

Does not generate

It was getting louder and louder.

Does not generate source

When did you visit New York?

Properties of error mining tasks:

	Parsing	Generation
Input	unannotated corpora	
Output	N-grams of input	N-grams of grammar rules
Common Errors Found	lexical inadequacies, MWEs, Unhandled constructs	Incorrect constructs, missing trigger rules, Overgeneration

Conclusion

Future Work

- Fine tune rule selection to prevent similar paths
- Add more characteristics (performance related, more robust MRS comparison)
- Replace lex-ids with lexical types
- Compare with more than top parse
- Fully automated
- Integrated in TSDB?
- Parsing Errors (perhaps with partial parses?)

Francis Bond, Eric Nichols, Darren Scott Appling, and Michael Paul. 2008. Improving statistical machine translation by paraphrasing the training data. In *International Workshop on Spoken Language Translation*, pages 150–157. Honolulu.

Daniël de Kok, Jianqiang Ma, and Gertjan van Noord. 2009. A generalized method for iterative error mining in parsing results. In *Grammar Engineering Across Frameworks (GEAF 2009)*.

Dan Flickinger. 2008. The English resource grammar. Technical Report 2007-7, LOGON, `http://www.emmtee.net/reports/7.pdf`. (Draft of 2008-11-30).

Benoît. Sagot and Éric. de La Clergerie. 2006. Error mining in parsing results. In *Annual Meeting-Association For Computational Linguistics*, volume 44, page 329.

Gertjan van Noord. 2004. Error mining for wide-coverage grammar engineering. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics Morristown, NJ, USA.