Machine Translation in MOLTO

Aarne Ranta

GF Meets SMT, Gothenburg, 1-5 November 2010

Multilingual Online Translation, FP7-ICT-247914

`www.molto-project.eu`

Beginnings of machine translation

Weaver 1947, encouraged by cryptography in WW II

Word lookup $\longrightarrow$ n-gram models (Shannon's "noisy channel")

```
ê = argmax P(f|e)P(e)
       e
```

P(w1 ... wn) approximated by e.g. P(w1w2)P(w2w3)...P(w(n-1)wn) (2-grams)

Word sense disambiguation

Eng. *even* $\longrightarrow$ Fre *égal*, *équitable*, *pair*, *plat* ; *même*, ...

Eng. *even number* $\longrightarrow$ Fre *nombre pair*

Eng. *not even* $\longrightarrow$ Fre *même pas*

Eng. *7 is not even* $\longrightarrow$ Fre *7 n'est pas pair*

Long-distance dependencies

Ger. *er bringt mich um* $\longrightarrow$ Eng. *he kills me*

Ger. *er bringt seinen besten Freund um* $\longrightarrow$ Eng. *he kills his best friend*

Bar-Hillel's criticism

1963: FAHQT (Fully Automatic High-Quality Translation) is impossible - not only in foreseeable future but in principle.

Example: word sense disambiguation for *pen*:

the pen is in the box vs. *the box is in the pen*

Requires unlimited intelligence, universal encyclopedia.

Trade-off: **coverage** vs. **precision**

The ALPAC report

Automatic Language Processing Advisory Committee, 1966

Conclusion: MT funding had been wasted money

Outcome: MT changed to more modest goals of *computational linguistics*: to describe language

Main criticisms: MT was too expensive

- too much postprocessing needed
- only small needs for translation - well covered by humans

1970's and 1980's

Movement from coverage to precision

Precision-oriented systems: Curry $\longrightarrow$ Montague $\longrightarrow$ Rosetta

Interactive systems (Kay 1979/1996)

- ask for disambiguation if necessary
- text editor + translation memory

Present day

IBM system (Brown, Jelinek, & al. 1990): back to Shannon's model

Google translate 2007- (Och, Ney, Koehn, ...)

- 57 languages

- models built automatically from text data

Browsing quality rather than *publication quality*

(Systran/Babelfish: rule-based, since 1960's)

The MOLTO project

Multilingual On-Line Translation

FP7-ICT-247914

Mission: to develop a set of tools for translating texts between *multiple languages* in *real time* with *high quality*.

`www.molto-project.eu`

Consumer vs. producer quality

Tool	**Google, Babelfish**	**MOLTO**
target	consumers	producers
input	unpredictable	predictable
coverage	unlimited	limited
quality	browsing	publishing

Producer's quality

Cannot afford translating

- *prix 99 euros*

to

- *pris 99 kronor*

Producer's quality

Cannot afford translating

- *I miss her*

to

- *je m'ennuie d'elle*

  ("I'm bored of her")

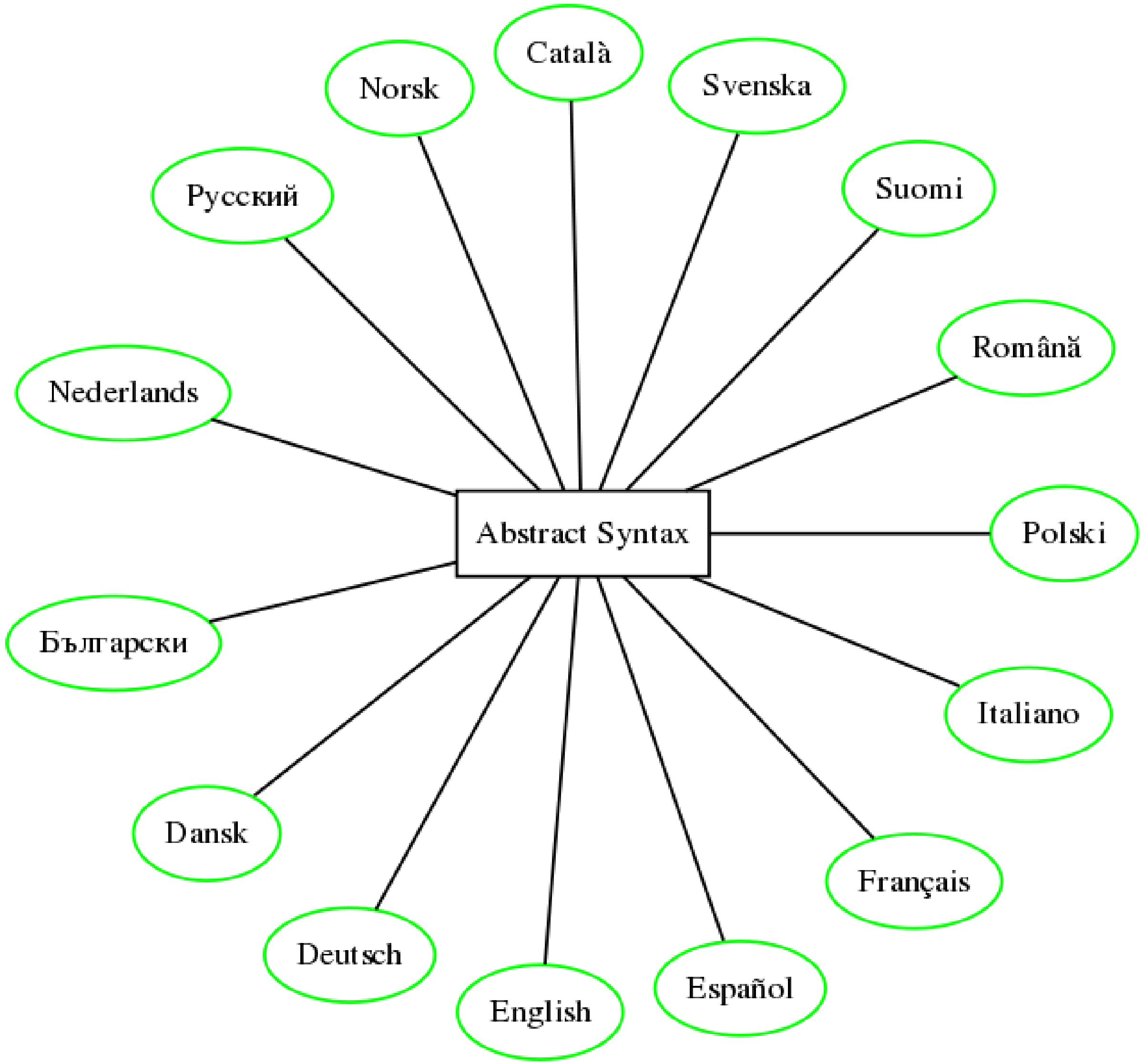
The translation directions

Statistical methods (e.g. Google translate) work decently *to* English

- rigid word order
- simple morphology
- focus of research funded by U.S. defence

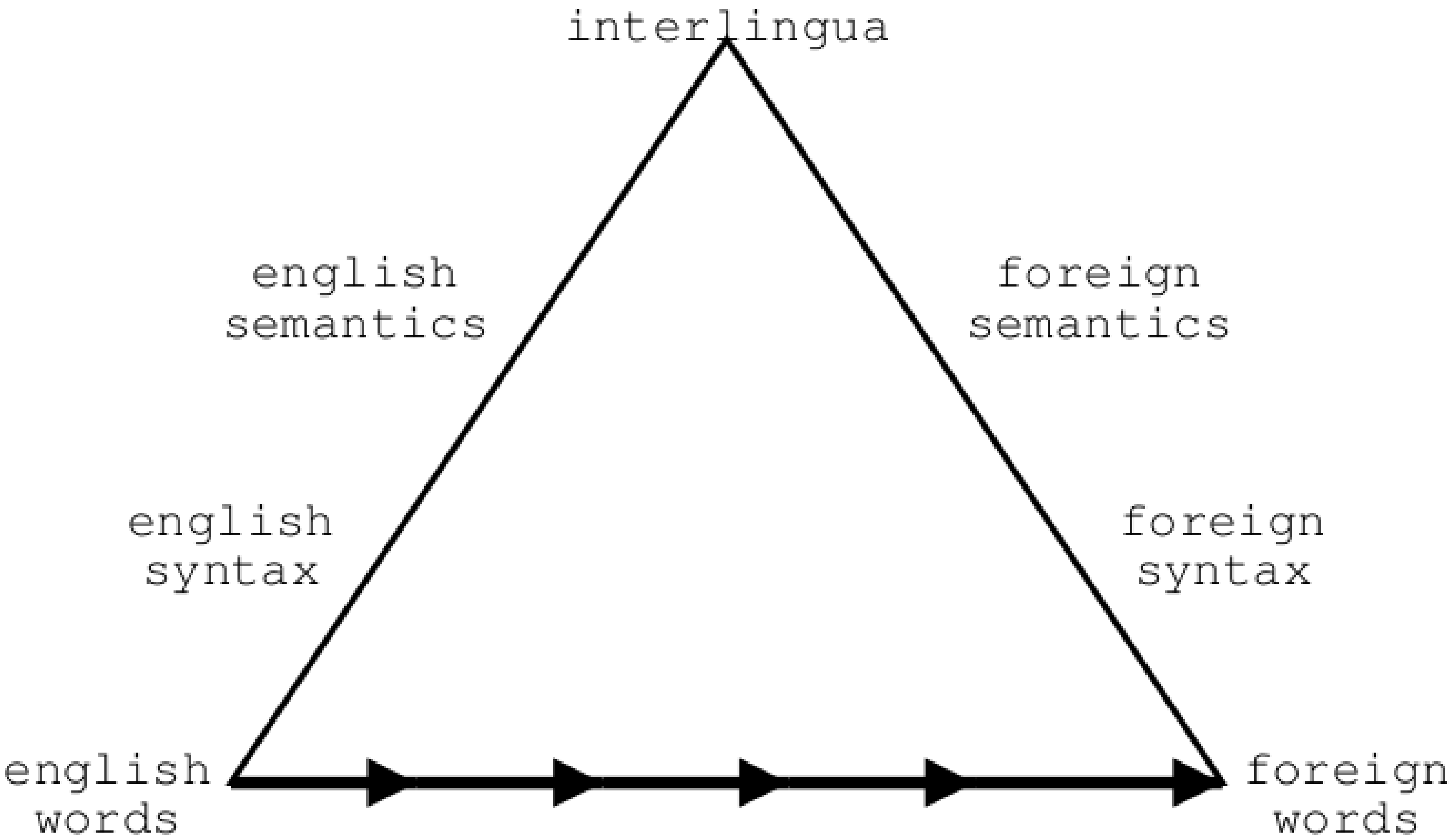
Grammar-based methods work equally well for different languages

- Finnish cases, German word order



MOLTO languages

The Vauquois Triangle



(From Knight & Koehn 2003)

Examples of translation levels

there are five gay bars in this street

```
            meaning
              /\              det finns fem gay-barer på denna gata
             /  \
            /    \
           /      \
          /        \          det finns fem glada stänger på denna gata
         /          \
        /            \        där är fem glada stänger i denna gata
       /              \
      /________________\      där är fem glad stång i detta gata
  Eng                    Swe
```

The fundamental problem with interlingua

[an interlingua for translation should] establish an order among all thoughts that can enter in the human spirit, in the same way as there is a natural order among numbers, and as one can learn in one day the names of all numbers up to infinity and write them in an unknown language, even though they are an infinity of different words...

The invention of this language depends on the true philosophy; for it is impossible otherwise to denumerate all thoughts of men and order them, or even distinguish them into clear and simple ones...

(Descartes, letter to Mersenne 1629)

Domain-specific interlinguas

The abstract syntax must be formally specified, well-understood

- semantic model for translation
- fixed word senses
- proper idioms

Examples of domain semantics

Expressed in various formal languages

- mathematics, in predicate logic
- software functionality, in UML/OCL
- dialogue system actions, in SISR
- museum object descriptions, in OWL

Type theory can be used for any of these!

Two things we do better than before

No universal interlingua:

- *The Rosetta stone is not a monolith, but a boulder field.*

Yes universal concrete syntax:

- no hand-crafted *ad hoc* grammars
- but a general-purpose resource grammar library

Grammatical Framework (GF)

Background: type theory, logical frameworks (LF)

GF = LF + concrete syntax

Started at Xerox (XRCE Grenoble) in 1998 for **multilingual document authoring**

Functional language with dependent types, parametrized modules, optimizing compiler

Factoring out functionalities

GF grammars are declarative programs that define

- parsing
- generation
- translation
- editing

Some of this can also be found in BNF/Yacc, HPSG/LKB, LFG/XLE ...

Multilingual grammars in compilers

Source and target language related by abstract syntax

```
                                                      iconst_2
                                                      iload_0
2 * x + 1  <----->  plus (times 2 x) 1  <------>      imul
                                                      iconst_1
                                                      iadd
```

A GF grammar for expressions

```
abstract Expr = {
  cat Exp ;
  fun plus : Exp -> Exp -> Exp ;
  fun times : Exp -> Exp -> Exp ;
  fun one, two : Exp ;
  }
```

```
concrete ExprJava of Expr = {
  lincat Exp = Str ;
  lin plus x y = x ++ "+" ++ y ;
  lin times x y = x ++ "*" ++ y ;
  lin one = "1" ;
  lin two = "2" ;
  }
```

```
concrete ExprJVM of Expr= {
  lincat Expr = Str ;
  lin plus x y = x ++ y ++ "iadd" ;
  lin times x y = x ++ y ++ "imul" ;
  lin one = "iconst_1" ;
  lin two = "iconst_2" ;
  }
```

Example: social network

Abstract syntax:

```
cat Message ; Person ; Item ;
fun Like : Person -> Item -> Message ;
```

Concrete syntax (first approximation):

```
lin Like x y = x ++ "likes" ++ y      -- Eng
lin Like x y = x ++ "tycker om" ++ y -- Swe
lin Like x y = y ++ "piace a" ++ x   -- Ita
```

Complexity of concrete syntax

Italian: agreement, rection, clitics (*il vino piace a Maria* vs. *il vino mi piace* ; *tu mi piaci*)

```
lin Like x y = y.s ! nominative ++ case x.isPron of {
  True  => x.s ! dative ++ piacere_V ! y.agr ;
  False => piacere_V ! y.agr ++ "a" ++ x.s ! accusative
  }
oper piacere_V = verbForms "piaccio" "piaci" "piace" ...
```

Moreover: contractions (*tu piaci ai bambini*), tenses, mood, ...

The GF Resource Grammar Library

Currently for 16 languages; 3-6 months for a new language.

Complete morphology, comprehensive syntax, lexicon of irregular words.

Common syntax API:

```
lin Like x y = mkCl x (mkV2 (mkV "like")) y         -- Eng
lin Like x y = mkCl x (mkV2 (mkV "tycker") "om") y -- Swe
lin Like x y = mkCl y (mkV2 piacere_V dative) x    -- Ita
```

Example-based grammar writing

Abstract syntax	`Like She He`	first grammarian
English example	*she likes him*	first grammarian
German translation	*er gefällt ihr*	**human translator**
resource tree	`mkCl he_Pron gefallen_V2 she_Pron`	GF parser
concrete syntax rule	`Like x y = mkCl y gefallen_V2 x`	variables renamed

GF meets SMT

1. Statistical Machine Translation (SMT) as fall-back

2. Hybrid systems

3. Learning of GF grammars by statistics

4. Improving SMT by grammars

Learning GF grammars by statistics

Abstract syntax	`Like She He`	first grammarian
English example	*she likes him*	first grammarian
German translation	*er gefällt ihr*	**SMT system**
resource tree	`mkCl he_Pron gefallen_V2 she_Pron`	GF parser
concrete syntax rule	`Like x y = mkCl y gefallen_V2 x`	variables renamed

Rationale: SMT is *good* for sentences that are *short* and *frequent*

Improving SMT by grammars

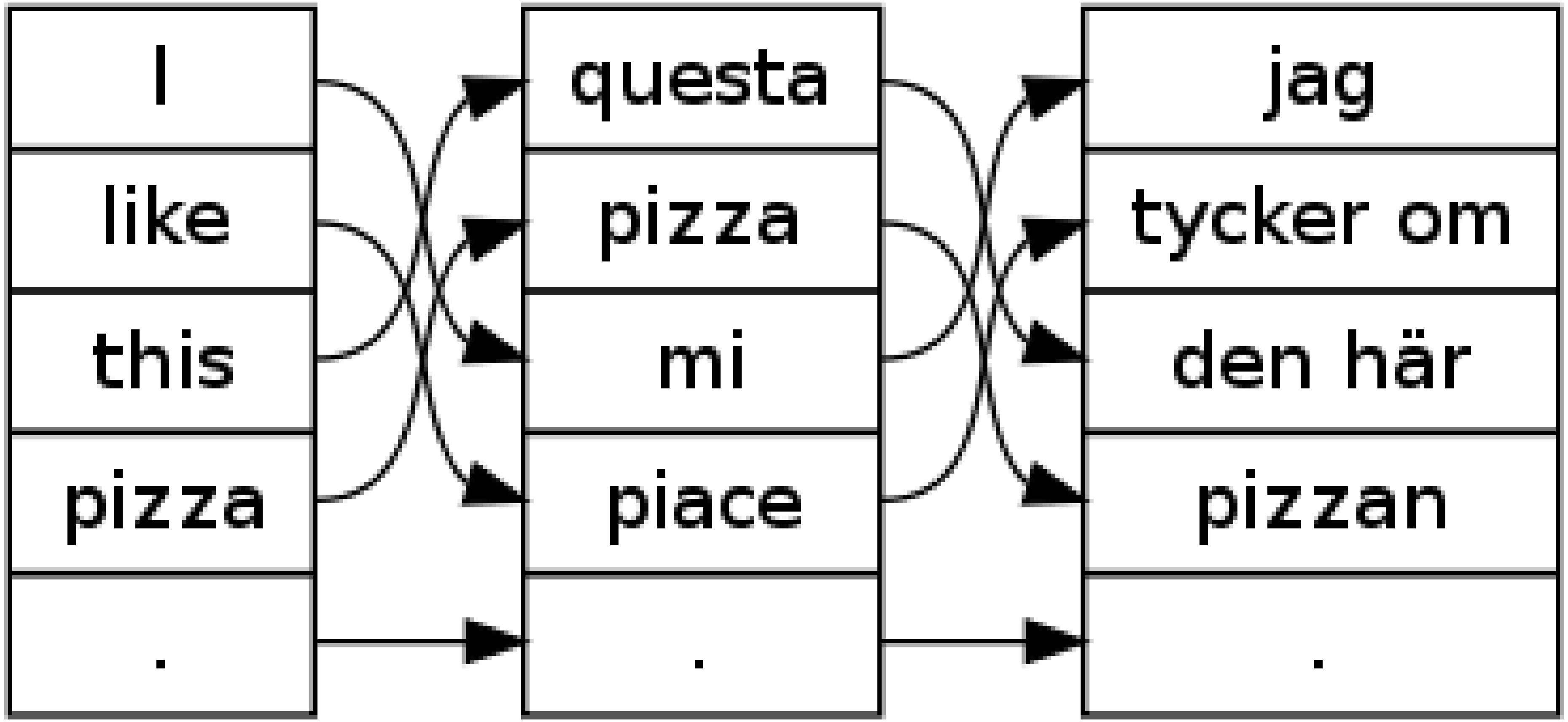
Rationale: SMT is *bad* for sentences that are *long* and involve *word order variations*

if you like me, I like you

`If (Like You I) (Like I You)`

wenn ich dir gefalle, gefällst du mir

Word/phrase alignments via abstract syntax



From grammar to SMT model

1. Generate bilingual corpus and word alignments from grammar

   - reliable alignments

   - good coverage of word forms and combinations

   - (however, unnatural distributions)

2. Use the resulting SMT model as fall-back for grammar-based translation

One scenario

```
SMT model 1
    |         resource grammar
    v
GF grammar
    |         corpus generation
    v
SMT model 2
```

Linguistic information in SMT

Factored models: replace bare word forms by lemma + analysis

Synchronous grammars: S-CFG, S-TAG, S-PMCFG ($\approx$ PGF)

Word-sense disambiguation

Additional features

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism
+ predictability	- unpredictability

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism
+ predictability	- unpredictability
- human effort	+ automatic production

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism
+ predictability	- unpredictability
- human effort	+ automatic production
- knowledge-intensive	+ data-driven

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism
+ predictability	- unpredictability
- human effort	+ automatic production
- knowledge-intensive	+ data-driven
- brittleness	+ robustness

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism
+ predictability	- unpredictability
- human effort	+ automatic production
- knowledge-intensive	+ data-driven
- brittleness	+ robustness
- human error risk	+ fidelity to data

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism
+ predictability	- unpredictability
- human effort	+ automatic production
- knowledge-intensive	+ data-driven
- brittleness	+ robustness
- human error risk	+ fidelity to data
- byzantine constructs	+ fluency

Grammars vs. SMT: pros and cons

Grammars	SMT
+ grammaticality	- word salad
+ long-distance dep's	- just local dep's
+ generality over data	- sparse data problem
+ modularity	- mix of levels
+ programmability	- holism
+ predictability	- unpredictability
- human effort	+ automatic production
- knowledge-intensive	+ data-driven
- brittleness	+ robustness
- human error risk	+ fidelity to data
- byzantine constructs	+ fluency
- so far only in small scale	+ exists in large scale

A word of wisdom on grammar vs. statistics

Grammar: **structures** of data

Statistics: **distribution** of data

These are orthogonal issues!

(Thanks: Gérard Huet)