

# Topics in Computational Linguistics — Grammar Engineering —

**Dan Flickinger**

CSLI Stanford & Saarland University

`danf@csli.stanford.edu`

**Stephan Oepen**

Universitetet i Oslo & CSLI Stanford

`oe@csli.stanford.edu`

<http://lingo.stanford.edu/courses/05/ge/>

# Recap: The Georgetown Experiment (1954)

*Russian is Turned into English by a Fast Electronic Translator*

(NYT, January 8, 1954)

*The switch is assured in advance by attaching the rule sign 21 to the Russian 'gyeneral' in the bilingual glossary which is stored in the machine, and by attaching the rule-sign 110 to the Russian 'mayor'. The stored instructions, along with the glossary, say whenever you read a rule sign 110 in the glossary, go back and look for a rule-sign 21. If you find 21, print the two words that follow it in reverse order.*

(Journal of Franklin Institute, March 1954)

- First public MT demonstration: IBM – GU cooperation (Lèon Dostert);
- somewhat limited scale: 250 words, six 'syntax' rules → high hopes.



# Machine Translation — Fundamentals

- Fully or partly automated translation: source to target language(s);
- MT for assimilation: rough translation to assess gist of documents;  
→ various three-letter agencies; on-line web translation (Systran);
- MT for dissemination: aim for quality resembling human translators;  
→ multi-lingual communities (EU, Canada); technical documentation;
- varying degree of abstraction: direct adaptation, transfer, interlingua.

*Come you with morning-boat, start tour-the same day-the.*

- Non-lexical content; interpretation to large part determined by syntax;
- translational equivalence not always determined by truth conditions.



# Machine Translation: Vauquois Triangle



---

STANFORD — 1-MAR-05 (oe@csli.stanford.edu)

Grammar Engineering (104)

# 'Realistic' Vauquois Triangle (Copestake, 1993)



---

STANFORD — 1-MAR-05 (oe@csli.stanford.edu)

Grammar Engineering (105)

# Some More Examples

*Snakker du norsk — Speak you Norwegian*

*Han synger gjerne — He sings with pleasure*

*å krysse fjorden svømmende — to cross fjord-the swimming*

*Stien følges nordover — Path-the is followed northwards*

*fetter | kusine — cousin*

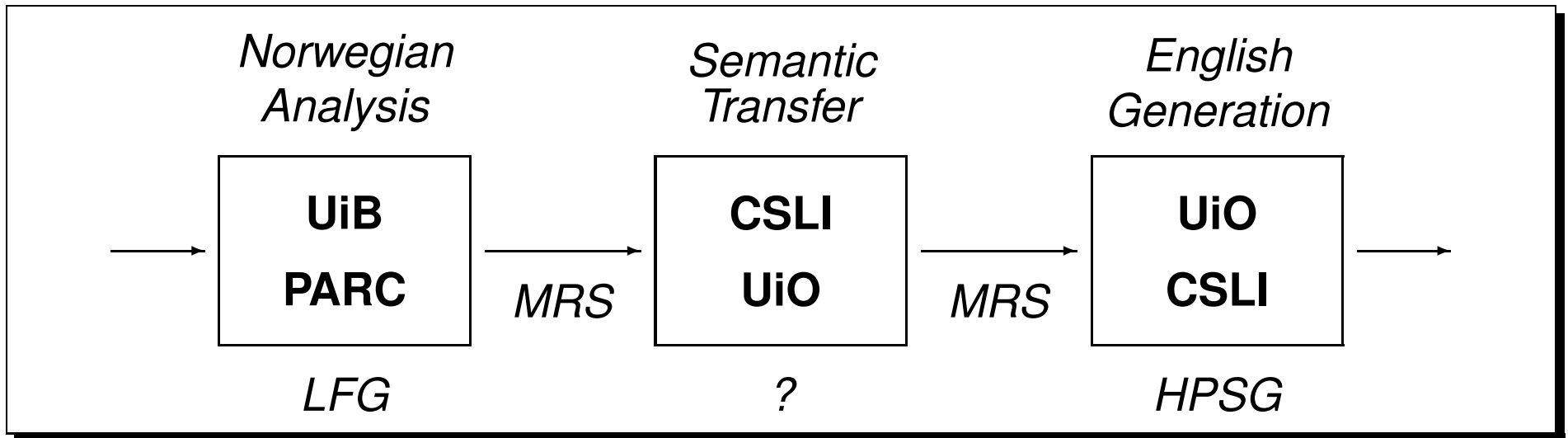
*bror — older | younger brother — distinct lexemes in Japanese*

## Semantic Transfer

- SL to TL mapping on reasonably abstract semantic representations;
- SL analysis using general grammar: syntax – semantics interface;
- grammar-based TL generation ‘guarantees’ grammatical output.



# Schematic MT Architecture



## Some LOGON Highlights

- Abstract from grammar idiosyncrasies: Minimal Recursion Semantics;
  - stochastic processes for disambiguation at all levels (and end-to-end);
- most likely unique in combination of LFG and HPSG in working system.



# Minimal Recursion Semantics — Essentials

$$\left[ \begin{array}{l} \text{TOP } h_1 \\ \text{INDEX } e_2 \\ \text{RELS } \left\langle \begin{array}{l} \left[ \begin{array}{l} \textit{prpstn\_m\_rel} \\ \text{LBL } h_1 \\ \text{MARG } h_3 \end{array} \right] \left[ \begin{array}{l} \textit{\_the\_q\_rel} \\ \text{LBL } h_4 \\ \text{ARGO } x_5 \\ \text{RSTR } h_6 \\ \text{BODY } h_7 \end{array} \right] \left[ \begin{array}{l} \textit{\_city\_n\_rel} \\ \text{LBL } h_8 \\ \text{ARGO } x_5 \end{array} \right] \left[ \begin{array}{l} \textit{\_populate\_v\_rel} \\ \text{LBL } h_9 \\ \text{ARGO } e_2 \\ \text{ARG1 } u_9 \\ \text{ARG2 } x_5 \end{array} \right] \left[ \begin{array}{l} \textit{\_densely\_r\_rel} \\ \text{LBL } h_{10} \\ \text{ARGO } e_{11} \\ \text{ARG1 } e_2 \end{array} \right] \end{array} \right\rangle \\ \text{HCONS } \langle h_3 =_q h_9, h_6 =_q h_8 \rangle \end{array} \right]$$

- An MRS is (a) a bag (aka multi-set) of labeled *elementary predications*, the (b) *top handle*, (c) *top index*, and (d) a set of *handle constraints*;
- predicates have *role–argument* pairs; predicates & variables are typed;
- small set of uninterpreted roles with external ‘thematic role’ assignment;
- (Copestake et al. 2003) fundamentals; (Copestake et al. 2000) algebra.



# Semantic Transfer: Minimal Recursion Semantics

## Background

- Family of flat, underspecified semantics (including UDRS, CLLS, et al.);
- suitable for large-scale implementation in unification-based grammars;
- non-recursive composition: predicates, bindings, and scope constraints;
- originally proposed for (among others) MT, now first-time instantiation;
- large, MRS-enabled grammars available for at least seven languages.

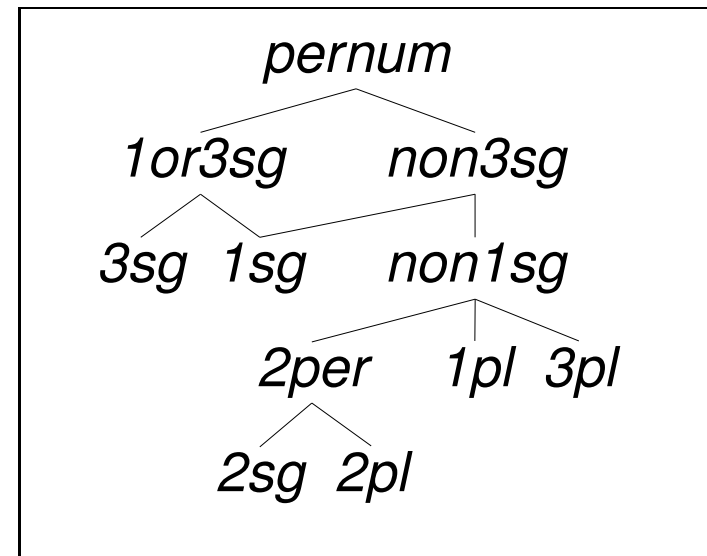
## Complex (Structured) Variables in MRSs

- Semantic agreement, (co-)referentiality, et al. by equation of *indices*;
- events: tense, aspect, mood (in terms of morpho-syntactic properties);
- referential indices: person, number, (natural) gender, and pronoun type.



# Index Properties in MRS — Examples

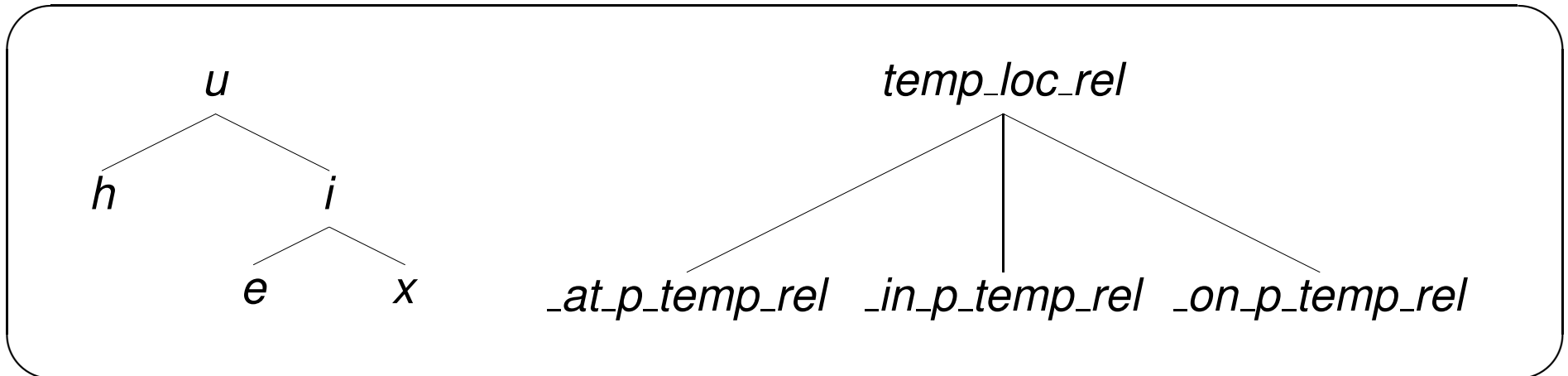
object := index &  
[ PN pernum,  
GEN gender ].



- Additional device of semantic (de-)composition); limited embedding;  
$$\text{feminine-rel}[\text{ARG0 } x_0] \wedge \text{parent-rel}[\text{ARG0 } x_0]$$
$$\longrightarrow \text{parent-rel}[\text{ARG0 } x_0 : \{ \text{GEN } f \} ]$$
- formally equivalent, but more or less convenient in interface to syntax.



# Use of Typing in MRSs



## Benefits of Typing

- variables and predicates are typed: generalizations at transfer level;
- appropriateness: inventory of properties and values for complex indices;
- predicate hierarchies facilitate underspecification in generator input;
- in transfer, use of types captures translational correspondence patterns.



# The LOGON Transfer Formalism

## Background

- *Semantic transfer* as successive rewriting of meaning representation;
- transfer rule: replacement of SL MRS fragment with TL correspondence;
- + complex, non-monotonic transformations; works well with *flat* structures;
- + good handle on most translational correspondences *and* divergences;
- resource-sensitive process: rule ordering; reversibility not guaranteed.

## Realization

- General-purpose, unification-based rewrite system on MRS structures;
  - allow non-determinism (ambiguity), but tight control in transfer grammar;
- sets of rules; manual ordering of rules and sets relative to each other.



# The Format of Transfer Rules

[ CONTEXT : ] INPUT [ ! FILTER ] → OUTPUT

- Transfer rules match one or more EPs and substitute their replacement;
- FILTER, CONTEXT, INPUT, and OUTPUT components are all partial MRSs;
- FILTER component blocks applicability of rules — a ‘negative’ match;
- variables bound in CONTEXT and INPUT available on right-hand side;
- the typical case: INPUT from source, OUTPUT from target language;
- all components can exploit MRS predicate and variable type hierarchy.

```
raste_v := mrs_transfer_rule &  
[ INPUT [ RELS < [ PRED "_raste_v_rel", LBL #h, ARG0 #e, ARG1 #x ] > ],  
  OUTPUT [ RELS < [ PRED "_rest_v_rel", LBL #h, ARG0 #e, ARG1 #x ] > ] ].
```



# Use of Typing in Transfer Grammar

'mrs.tdl'

```
mrs_transfer_rule := top &  
[ FILTER mrs,  
  CONTEXT mrs,  
  INPUT mrs,  
  OUTPUT mrs ].
```

```
arg1_v_mtr := mrs_transfer_rule &  
[ INPUT [ RELS < [ LBL #h1, ARGO #e1, ARG1 #x1 ] > ],  
  OUTPUT [ RELS < [ LBL #h1, ARGO #e1, ARG1 #x1 ] > ] ].
```

'verbs.mtr'

```
raste_v := arg1_v_mtr &  
[ INPUT [ RELS < [ PRED "_raste_v_rel" ] > ],  
  OUTPUT [ RELS < [ PRED "_rest_v_rel" ] > ] ].
```



# Ambiguity in Transfer Grammars (1 of 2)

```
reise_travel_v := arg1_v_omtr &  
[ INPUT [ RELS < [ PRED "_reise_v_rel" ] > ],  
  OUTPUT [ RELS < [ PRED "_travel_v_rel" ] > ] ].
```

```
reise_leave_v := arg1_v_mtr &  
[ INPUT [ RELS < [ PRED "_reise_v_rel" ] > ],  
  OUTPUT [ RELS < [ PRED "_leave_v_rel" ] > ] ].
```

## Control Mechanisms

- Rules can apply *optionally* — rewrite process forks (non-determinism);
  - the search space grows *exponentially*, need for control mechanism;
  - obligatory rules (the default) consume the intermediate MRS at the time;
- order matters: apply more ‘specific’ (conditioned) rules earlier in process.



# Ambiguity in Transfer Grammars (2 of 2)

```
quantifier_mtr := monotonic_mtr &  
[ INPUT.RELS < [ LBL #h1, ARGO #x2, RSTR #h3, BODY #h4 ] > ,  
  OUTPUT.RELS < [ LBL #h1, ARGO #x2, RSTR #h3, BODY #h4 ] > ] .
```

```
def_the_q := quantifier_mtr &  
[ INPUT.RELS < [ PRED "def_q_re" ] >  
  OUTPUT.RELS < [ PRED _the_q_rel ] > ] .
```

- Multiple occurrences of same predicate give rise to additional ambiguity;
  - $R_i$  may apply to (parts of)  $M_j$  multiple times; potentially order-sensitive;
  - more non-determinism: unfold ambiguity but ‘pack’ under equivalence;
- chart-like transfer, compare alternate MRSs in canonical *normal form*;
- ? more long-term, investigate genuine local ambiguity packing in MRSs.



# Some LOGON Sample Data (Analyzed by LinGO ERG)

- 1 *Be considerate of game, farm animals and other hikers.*
- 109 *Kjeragveggen has interested climbers since the 1970s.*
- 304 *But there are things to do for those with knickers and anoraks too.*
- 39 *Follow the road past NUTEC and continue up along Kvarvenveien, past the recreation area.*
- 41 *Keep the height, walk around Bråkdalen, and soon you will stand on the top of Vassberget.*
- 248 *The first part of the trip goes with the Hurtigruta to Torvik, with a bicycle ride at night into the sunrise out to Runde, and a hike to Norway's southernmost bird mountain.*
- 326 *If there is one thing Swedes are concerned with, it is preparing delicious dishes.*



# The LinGO English Resource Grammar (ERG)

## Development Background (1993 – today)

- General-purpose, wide-coverage, computational English grammar;
- mainly Dan Flickinger, with Rob Malouf, Emily M. Bender, Jeff Smith;
- supported in multiple HPSG processing environments (LKB & PET);
- range of projects: speech, email, UseNet (no WSJ); one company.

## Design

- HPSG [Pollard & Sag 1994]: constraint-based, strongly lexicalized;
- MRS [Copestake et al., 1999]: flat, Davidsonian, underspecified;
- type hierarchies defining principles, lexical classes, constructions;
- strict grammaticality assumption: generator using same grammar.



# LinGO ERG: Coverage and Size

## Linguistic Coverage

- 85 % of 12,000 transcribed dialogue turns from VerbMobil domains;
- 80<sup>+</sup> % of customer emails in financial and ecommerce domains;
- both fairly short utterances: average 9 words, ranging from 1 – 40;
- 88 % of phenomena-based examples in Hewlett Packard test suite.;
- more recently, 95 % on excerpts from tourism brochures (13 words).

## Size of Grammar (as of October 2004)

- some 2,600 types for fundamentals, lexicon, rules, and semantics;
- 12,358 lexical entry stems (around 3,800 verbs and 6,300 nouns);
- 27 lexical (15 inflectional) rules and 96 phrase structure schemata.



# Grammatical Coverage on Development Corpus

'lingo/08-nov-03/hike/03-11-14/pet'						
Aggregate	total items	word string	lexical items	parser analyses	overall coverage	time (s)
	#	$\phi$	$\phi$	$\phi$	%	$\phi$
$35 \leq i\text{-length} < 40$	1	35.00	109.00	2372.00	100.0	2.00
$30 \leq i\text{-length} < 35$	2	32.50	109.00	1768.00	100.0	1.12
$25 \leq i\text{-length} < 30$	7	26.71	100.57	1393.14	100.0	1.32
$20 \leq i\text{-length} < 25$	28	21.68	78.36	931.93	100.0	0.52
$15 \leq i\text{-length} < 20$	72	16.89	54.08	136.18	93.1	0.26
$10 \leq i\text{-length} < 15$	119	11.77	39.85	35.87	95.0	0.10
$5 \leq i\text{-length} < 10$	95	7.47	23.49	5.79	93.7	0.04
$0 \leq i\text{-length} < 5$	6	4.00	7.67	1.33	100.0	0.01
<b>Total</b>	<b>330</b>	<b>12.86</b>	<b>42.85</b>	<b>177.17</b>	<b>94.8</b>	<b>0.19</b>

(generated by [incr tsdb()] at 14-nov-2003 (22:49 h))



# Why Ambiguity Management?

|< |Den andre bratte veien mot Bergen er kort.| (1) --- 6 x 48 x 144 = 144  
|> |the 2nd steep path towards bergen is a card| [717.7] <0.15> (0:4:0).  
|> |the second steep path towards bergen is a card| [801.9] <0.15> (0:4:1).  
|> |the other steep path towards bergen is a card| [821.2] <0.38> (1:4:0).  
|> |the steep 2nd road towards bergen is a card| [837.8] <0.32> (0:6:2).  
|> |the 2nd steep road towards bergen is a card| [923.7] <0.45> (0:6:0).  
|> |the second steep road towards bergen is a card| [1032.1] <0.45> (0:6:1).  
|> |this 2nd steep path towards bergen is a card| [1051.1] <0.13> (0:0:2).  
|> |the steep 2nd path towards bergen is a card| [1053.4] <0.15> (0:4:2).  
|> |the other steep road towards bergen is a card| [1056.9] <0.75> (1:6:0).  
|> |the 2nd steep path towards bergen is short| [1072.1] <0.37> (4:4:2).  
|> |the 2nd steep path against bergen is a card| [1097.1] <0.07> (0:5:0).  
|> |this steep 2nd road towards bergen is a card| [1131.9] <0.29> (0:2:0).  
|> |this second steep path towards bergen is a card| [1173.9] <0.13> (0:0:3).  
...  
|> |the other steep road towards bergen is short| [1657.0] <1.00> (5:6:0).  
...  
|> |this steep second path against bergen is cards| [11705.5] <0.07> (2:1:1).

