INF5820/INF9820

LANGUAGE TECHNOLOGICAL APPLICATIONS

Jan Tore Lønning, Lecture 8, 10 Oct. 2014
jtl@ifi.uio.no

Today

- Parameter tuning
- Reranking
- Hybrid translation
  - Rule-based backbone
  - Reranking
- A glimpse beyond

The generative SMT-model

- Adding weights:
  - Koehn, lecture 5, Slide 17-21

How to tune weights?

1. Make an original system, S0, using a parallel corpus, C1, for the phrase table.
2. Use a distinct small parallel corpus, C2. (dev set)
3. Produce several translations for each f-sentence in C2.
   - n-best list (n=100, 1000, 10000)
4. Use a method for scoring the candidate translations in C2.
   - (typically modified BLEU-score).
5. Try to adjust the weights to bring the best candidates in (4) towards top of list.
6. Make new system with adjusted weights.
7. Repeat from 3 towards convergence.

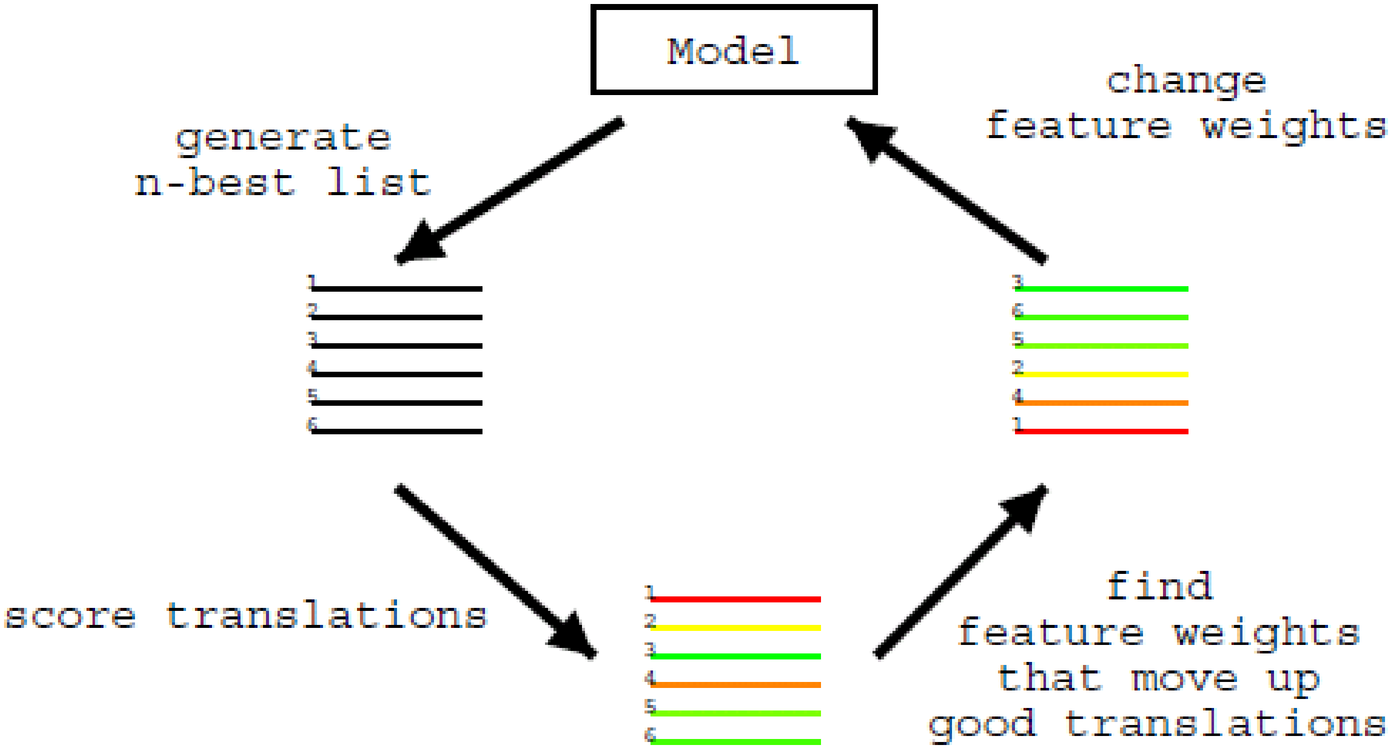
Learning task

- Task: *find weights*, so that feature vector of the correct translations *ranked first*

rank	TRANSLATION	LM	TM	WP	SER
1	Mary not give slap witch green .	-17.2	-5.2	-7	1
2	Mary not slap the witch green .	-16.3	-5.7	-7	1
3	Mary not give slap of the green witch .	-18.1	-4.9	-9	1
4	Mary not give of green witch .	-16.5	-5.1	-8	1
5	Mary did not slap the witch green .	-20.1	-4.7	-8	1
6	Mary did not slap green witch .	-15.5	-3.2	-7	1
7	Mary not slap of the witch green .	-19.2	-5.3	-8	1
8	Mary did not give slap of witch green .	-23.2	-5.0	-9	1
9	Mary did not give slap of the green witch .	-21.8	-4.4	-10	1
10	Mary did slap the witch green .	-15.5	-6.9	-7	1
11	**Mary did not slap the green witch .**	**-17.4**	**-5.3**	**-8**	**0**
12	Mary did slap witch green .	-16.9	-6.9	-6	1
13	Mary did slap the green witch .	-14.3	-7.1	-7	1
14	Mary did not slap the of green witch .	-24.2	-5.3	-9	1
15	Mary did not give slap the witch green .	-25.2	-5.5	-9	1

rank translation — feature vector

Discriminative training



How to? (sec. 9.3)

5. Try to adjust the weights to bring the best candidates in (4) towards top of list.

- No analytic solution
  - We can't differentiate a function and find zero values
- Take 1: try systematically, say
    - λ_{LM}= .1, .2, .3, …, .9
    - λ_{ϕ}= .1, .2, …, .9- λ_LM
    - λ_{D}= …
  - Too many values to try out
  - Small changes in λs, large effect on result:
    - The steps are too large

Take 2: Powell search

- Optimize one λ, say λ_{LM}, keeping the other fixed.
- With this value for λ_{LM}, optimize the next λ, etc.
- A method for searching for the best value for each λ

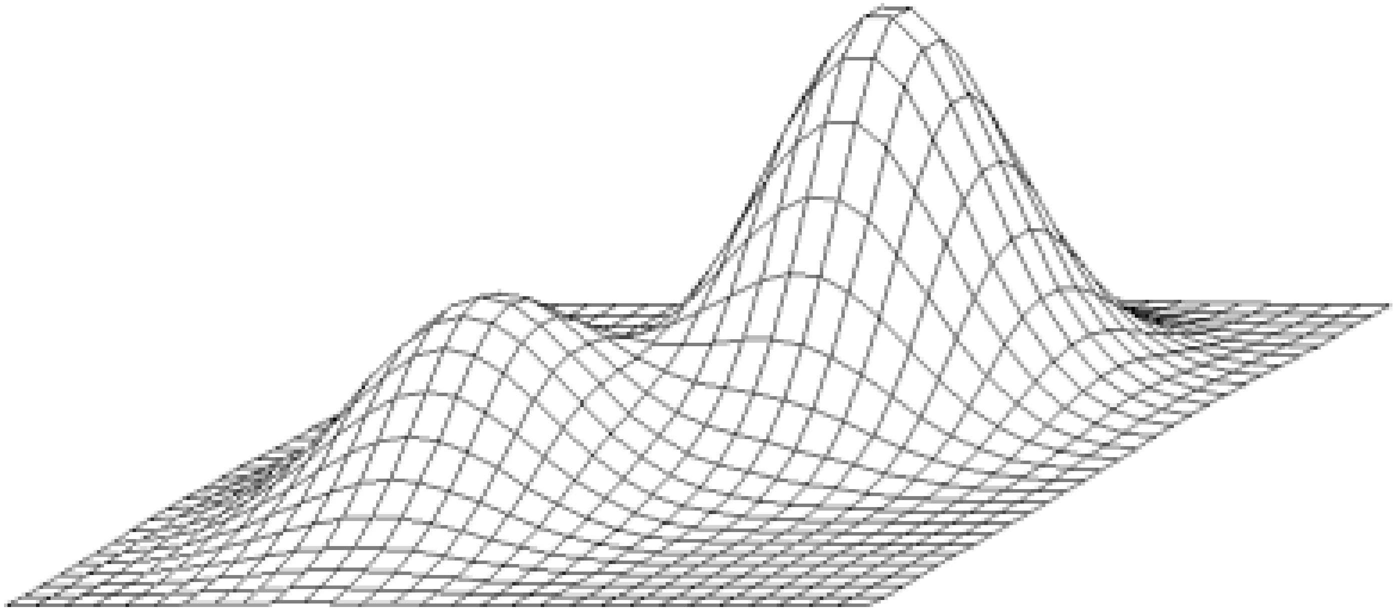
Take 3:

- (alternative) Simplex algorithm
- Variants of “hill climbing”

- Read sec 9.3
  - Not the details of
    - Finding threshold points
    - Combining threshold points

    in sec 9.3.2
  - Not 9.3.3 Simplex

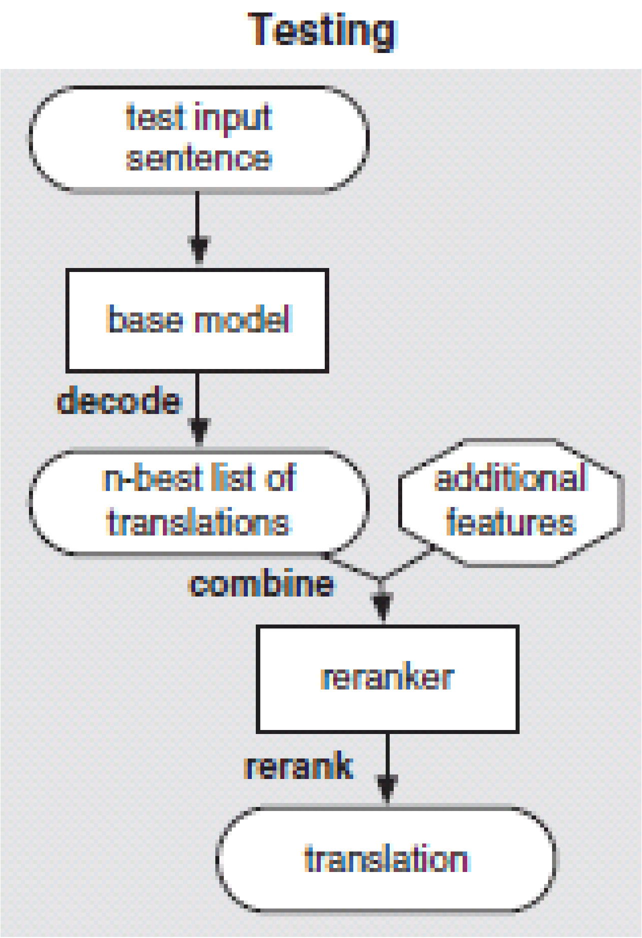
Will the solutions be global?



Today

- Parameter tuning
- Reranking
- Hybrid translation
  - Rule-based backbone
  - Reranking
- A glimpse beyond

Reranking model for SMT



- Sec. 9.2

Statistical models

Generative model	Discriminative model

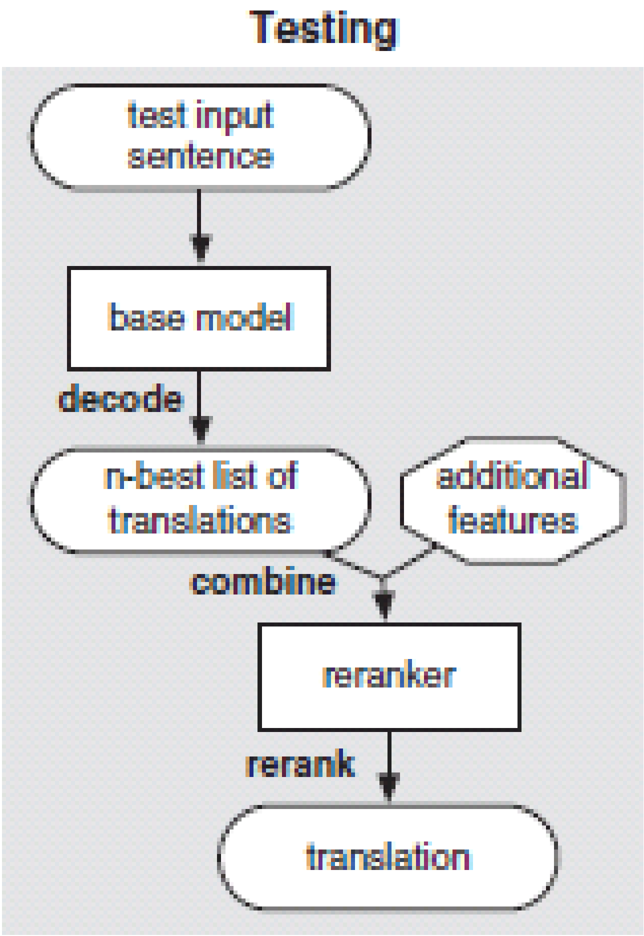
Generative model

- Construct solutions and assign them probabilities
- Examples
  - PCFG:
    - Assign trees
    - Probabilities to the trees
  - HMM-tagger
  - The translation models, both IBM and phrase-based

Discriminative model

- Starts with a set of solutions
- Select between them on the basis of a statistical score
- Example:
  - Malt parser

Reranking model for SMT



- Discriminative model
- Take as input an n-best list from a translation system

Reranking vs Tuning

- What is the difference between
  - Tuning and
  - Reranking?

Supervised learning

- Consider it as a classification problem
- Choose learning goal:
  - Typically modified BLEU (or NIST) score
- Choose features
- Alternative learning strategies:
  - Naïve Bayes
  - Maximum entropy
    - (INF5830)
    - Skip here 9.2.4
  - Etc.

A glimpse beyond

- Large-Scale (sec. 9.4 not to read)
- Millions of parameters
  - e.g. weight on each phrase probability
    - λ_345698 * P(the house | das haus)
    - λ_345699 * P(the building| das haus)
- Need large dev corpus for tuning

Today

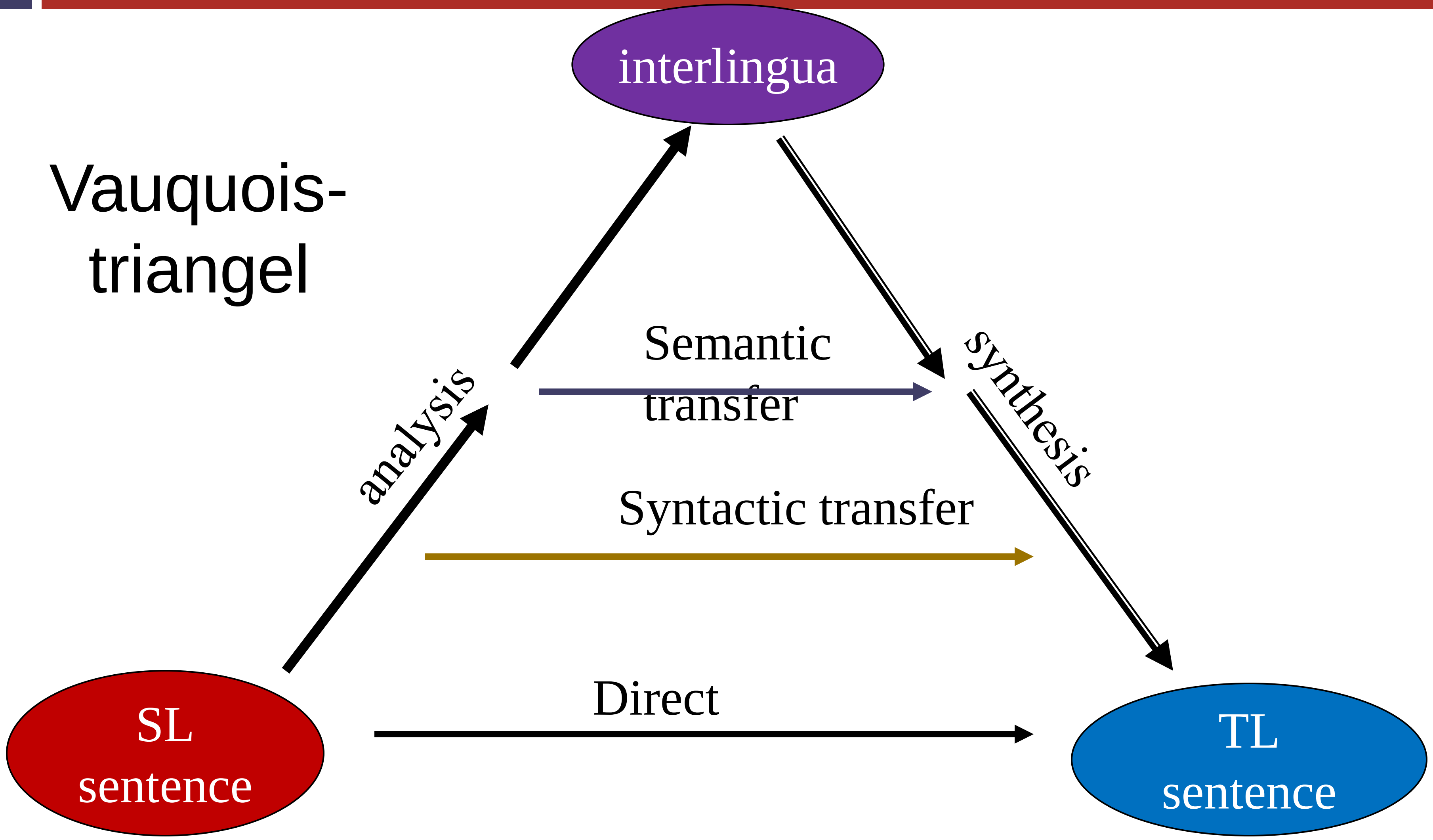
- Parameter tuning
- Reranking
- Hybrid translation
  - Rule-based backbone
  - Reranking
- A glimpse beyond

The LOGON project

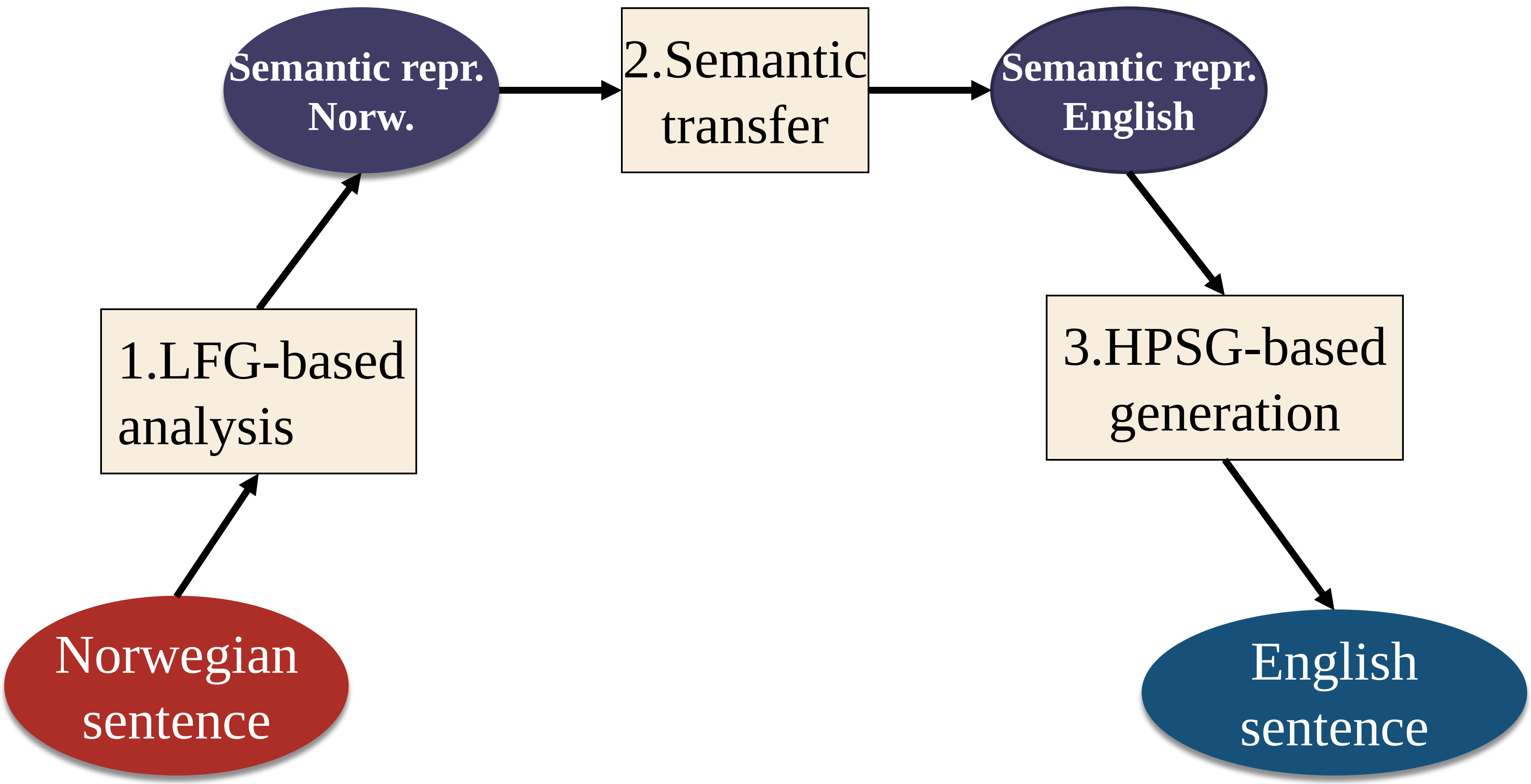
- MT: Norwegian → English
- Tourist texts – hiking descriptions
- High quality (precision) – limited recall
- 2003-2007
- Strategy
  - Mainly rule-based:
    - Semantic transfer
  - Statistical reranking

Alternative strategies

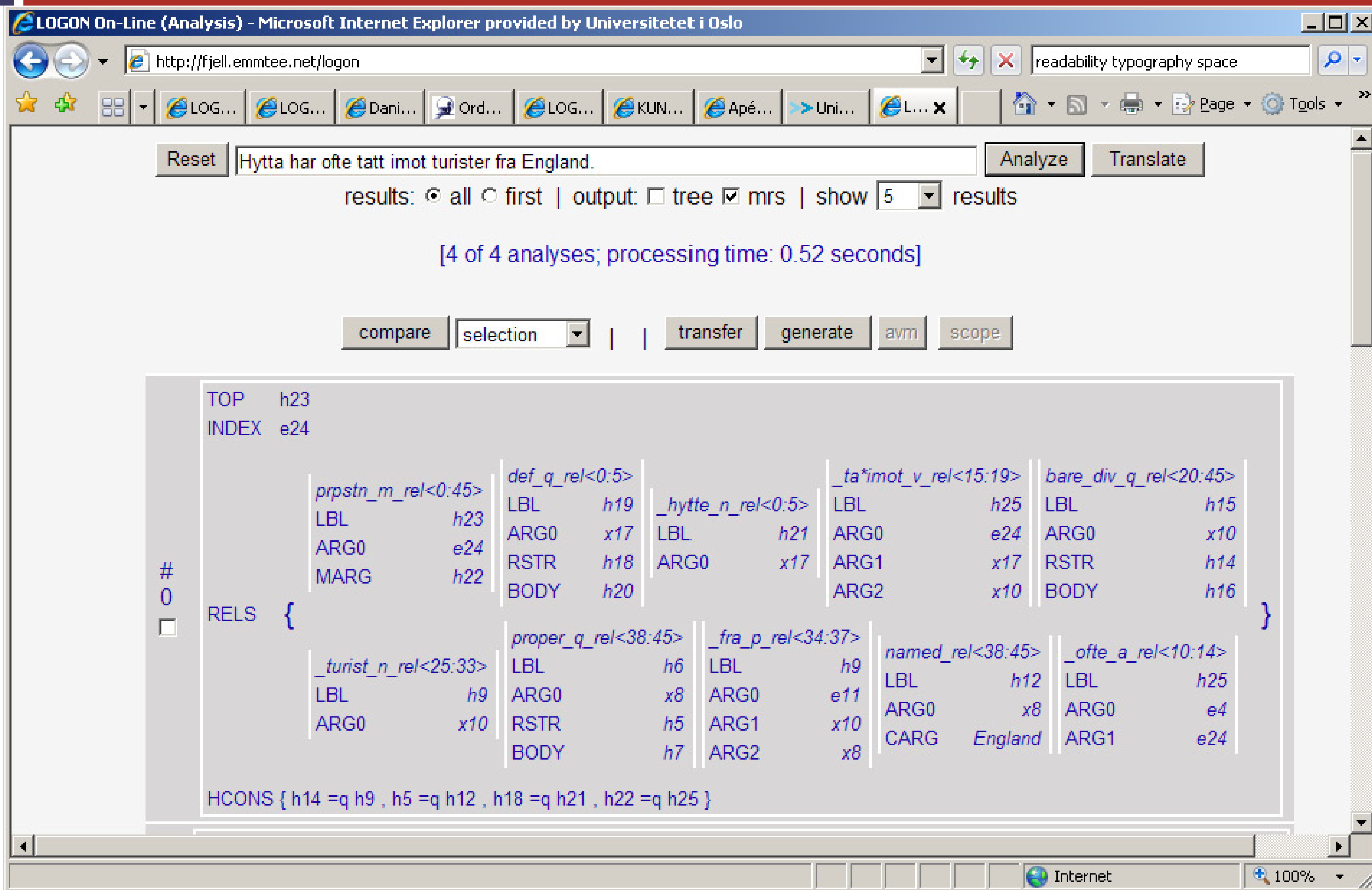
Vauquois-
triangel



Back bone: Semantic transfer



Minimal Recursion Semantics



Analysis

- Grammar: NorGram,
  - A multipurpose computational grammar based on LFG
  - Developed at UiB since 1998
  - LOGON has
    - greatly extended grammatical coverage
    - equipped it with an MRS semantics module
    - enhanced efficiency
- Processing
  - The XLE system from PARC
  - Morphological processing developed at UiB on top of earlier projects (tagging, UiB & UiO & NTNU)
  - Compositional analysis of compounds

Generation

- Grammar
  - The English Resource Grammar (ERG)
  - A multipurpose computational grammar based on HPSG
  - Continuously developed since 1994 (CSLI Stanford)
  - Refined, domain-adapted, and extended by LOGON
  - Open source, used in other ongoing projects
- Processing
  - Adapted technology from DELPH-IN consortium
  - LOGON: forty times faster generation algorithms

Transfer

- Grammar
  - Hand-coded transfer rules (7000 rules)
  - Semi-automatic acquisition of transfer correspondences
    - for open class words
    - from a dictionary (Kunnskapsforlagets store No-En)
    - (ca 10 000)
- Processing
  - Typed unification-based formalism for rewriting of MRSs
  - Design and implementation from scratch
  - Non-deterministic rewriting of MRS-fragments

Today

- Parameter tuning
- Reranking
- Hybrid translation
  - Rule-based backbone
  - Reranking
- A glimpse beyond

To be continued