Maximum Entropy Models for Realization Ranking

Erik Velldal[†]
<erik.velldal@iln.uio.no>

Stephan Oepen[†‡]
<oe@csli.stanford.edu>

[†] **Department of Linguistics and Scandinavian Studies, University of Oslo (Norway)**

[‡] **Center for the Study of Language and Information, Stanford (USA)**

Realization Ranking

- The problem: Ambiguity in generation, – many ways to formulate a given meaning.
- A solution: Use statistics for modeling preferences and *soft constraints* (grammaticality already guaranteed).
- Trained and tested three types of models:
  1) n-gram language models (surface oriented)
  2) maximum entropy model (structural features)
  3) a combination of 1) and 2)

Overview

- Generation in the LOGON MT-system and the problem of realization ranking.
- Reference experiments: random choice and n-gram language models.
- The relation to parse selection. Treebank data and maximum entropy models (MaxEnt).
- A combined model: MaxEnt + language model.
- Results, future work and discussion.

Generation in the LOGON MT-system

- LOGON
  - Aims at high precision Norwegian–English MT of texts in the tourism domain.
  - Symbolic, rule-based system, centered on semantic transfer using Minimal Recursion Semantics (MRS; Copestake, Flickinger, Malouf, Riehemann, & Sag, 1995). Includes stochastic methods for ambiguity management.

Generation in the LOGON MT-system

- LOGON
  - Aims at high precision Norwegian–English MT of texts in the tourism domain.
  - Symbolic, rule-based system, centered on semantic transfer using Minimal Recursion Semantics (MRS; Copestake, Flickinger, Malouf, Riehemann, & Sag, 1995). Includes stochastic methods for ambiguity management.
- The LKB Chart Generator (Carroll, Copestake, Flickinger, & Poznanski, 1999; Carroll & Oepen, 2005).
  - Lexically-driven, bottom-up chart generation from MRSs.
  - Generation based on the LinGO English Resource Grammar (ERG; Flickinger, 2002); a general-purpose, wide-coverage grammar, designed using HPSG and MRS.

Generator Ambiguity

- Caused by e.g. the optionality of complementizers and relative pronouns, permutation of (intersective) modifiers, different possible topicalizations, as well as lexical and orthographic alternations.
- Average number of realizations in the current data set is 73 (max = 5712). All realizations of a given MRS are guaranteed to be semantically (truth-conditionally) equivalent. Grammaticality is ensured wrt. the underlying grammar (LinGO ERG).

> *Remember that dogs must be on a leash.*
> *Remember dogs must be on a leash.*
> *On a leash remember that dogs must be.*
> *On a leash remember dogs must be.*
> *A leash remember that dogs must be on.*
> *A leash remember dogs must be on.*
> *Dogs remember must be on a leash.*

A Language Model Ranker

- The most common approach to the problem of generator ambiguity is to use n-gram statistics (Langkilde & Knight, 1998; White, 2004; Callison-Burch & Flournoy, 2001).
- Score and rank strings using a language model; $p_n(w_1, \ldots, w_k) = \prod_{i=1}^{k} p(w_i | w_{i-n}, \ldots, w_{i-1})$.
- Trained a 4-gram model on the BNC (100 mill. words).

A Language Model Ranker (Cont'd)

- Results on the LOGON data set 'Rondane' (864 test items, to be detailed later in the talk):
  - Exact match accuracy: 48.46%
    (Random choice baseline: 18.03%)
  - BLEU score: 0.8776
    (Random choice baseline: 0.727)
- Limitations: Can not model dependencies between non-contiguous words. No linguistic information. Does not condition the output (string) on the input (MRS).

The Relation to Parse Selection

- The problem of selecting the *best realization* can be seen to be "inversely similar" to the problem of selecting the *best parse*.
- $p(\text{analysis}|\text{utterance})$ *vs.* $p(\text{utterance}|\text{analysis})$

The Relation to Parse Selection

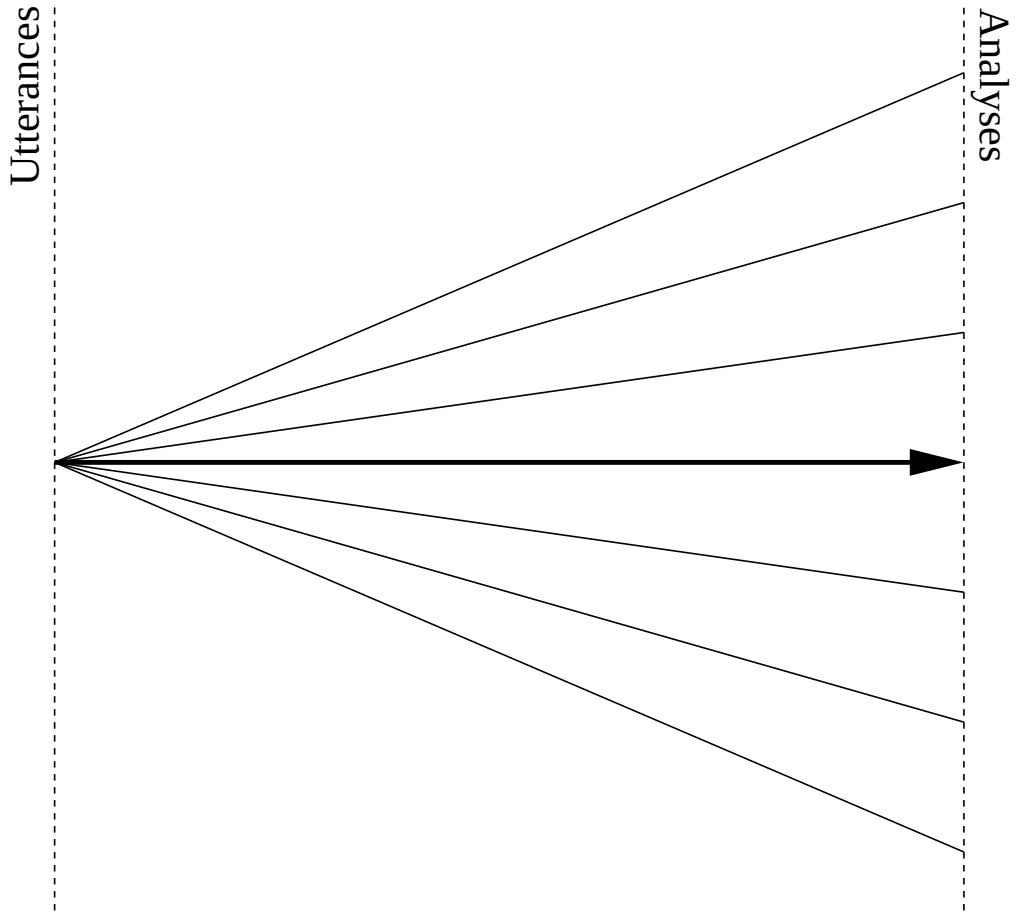
- The problem of selecting the *best realization* can be seen to be "inversely similar" to the problem of selecting the *best parse*.
- $p(\text{analysis}|\text{utterance})$ *vs.* $p(\text{utterance}|\text{analysis})$
- Toutanova, Manning, Shieber, Flickinger, & Oepen (2002) implement a MaxEnt model for *parse disambiguation* using the Redwoods HPSG treebank.
- Features defined over *derivation trees* with non-terminals representing the *construction types* and *lexical types* of the grammar.

The Relation to Parse Selection

- The problem of selecting the *best realization* can be seen to be "inversely similar" to the problem of selecting the *best parse*.
- $p(\text{analysis}|\text{utterance})$ *vs.* $p(\text{utterance}|\text{analysis})$
- Toutanova, Manning, Shieber, Flickinger, & Oepen (2002) implement a MaxEnt model for *parse disambiguation* using the Redwoods HPSG treebank.
- Features defined over *derivation trees* with non-terminals representing the *construction types* and *lexical types* of the grammar.
- We train a realization ranker in much the same way.
- Requires different types of treebanks for training.

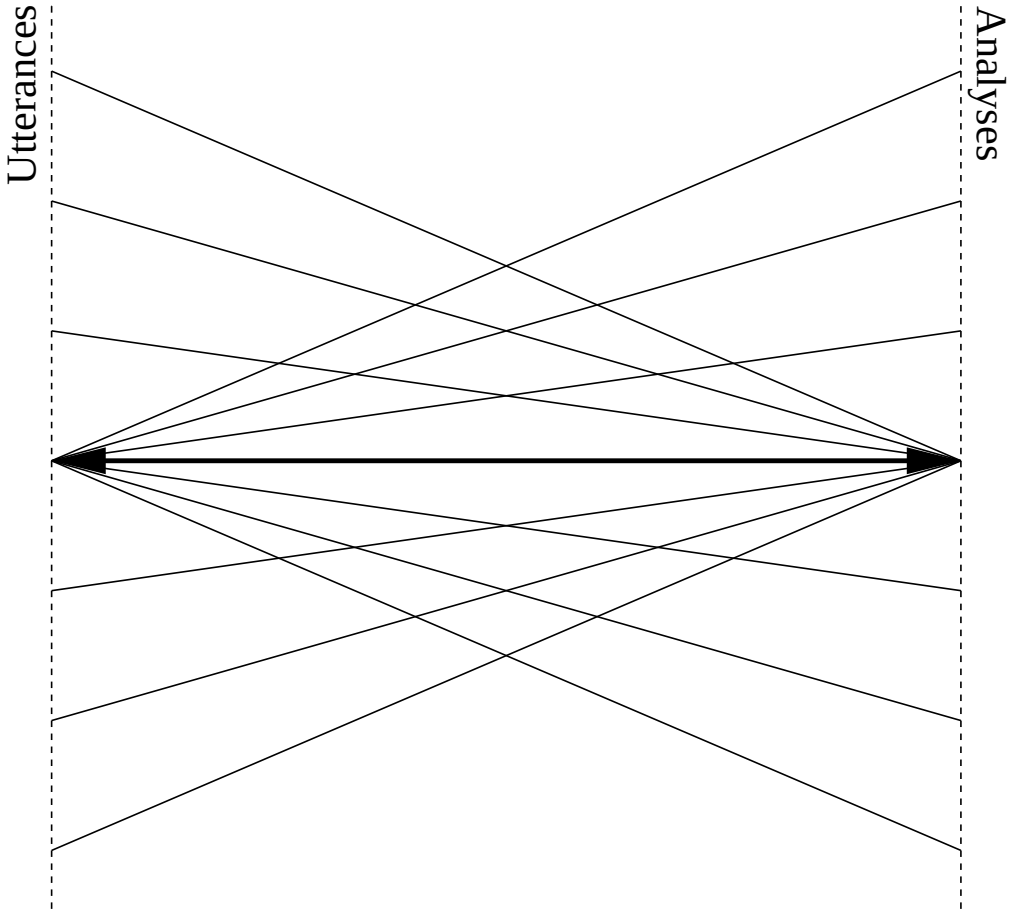
Treebanks for Parse Selection

Training data for parse selection models is typically given by (1) a treebank of utterances paired with their optimal analyses, together with (2) all their competing (sub-optimal) analyses.



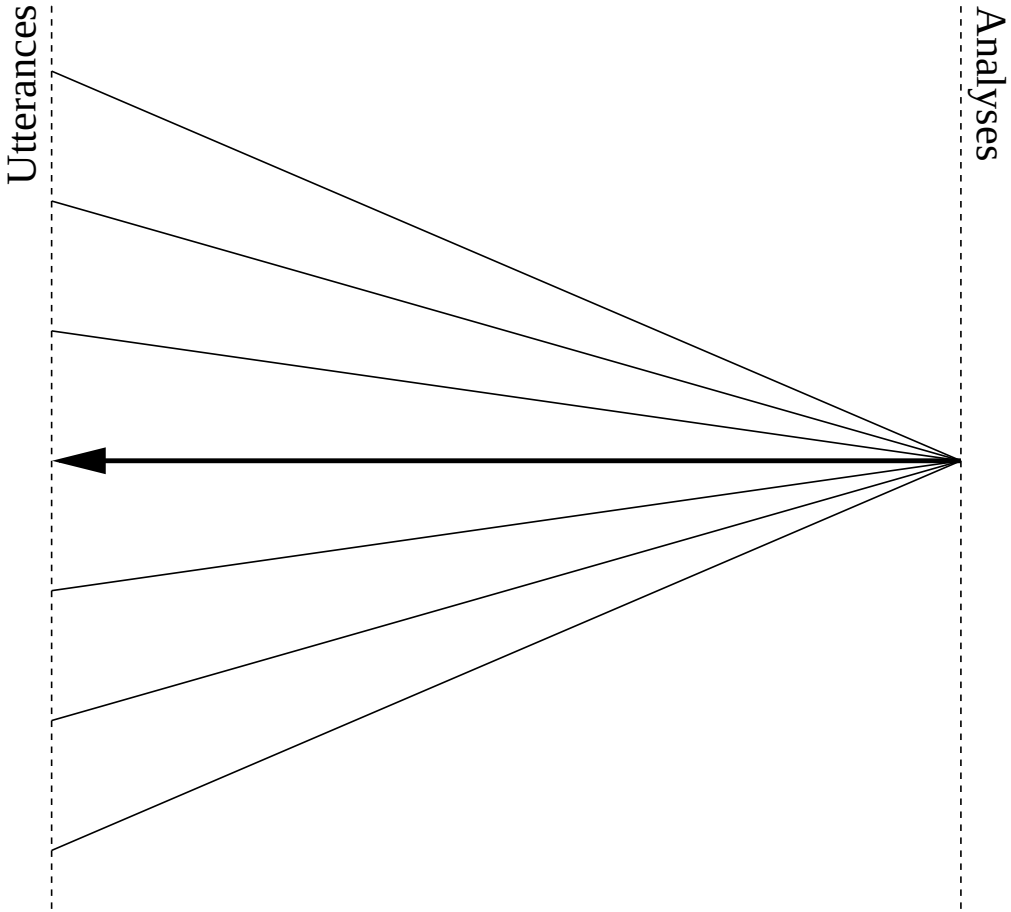
Symmetric Treebanks

To produce a *symmetric treebank*, exhaustively generate all paraphrases of the treebanked analyses, and assume optimality relation to be *bidirectional* (Velldal, Oepen, & Flickinger, 2004).



Treebanks for Realization Ranking

We now have the training data for a realization ranking model, given by (1) a treebank of analyses paired with their optimal utterances, together with (2) all competing (suboptimal) candidates.



The Rondane Treebank

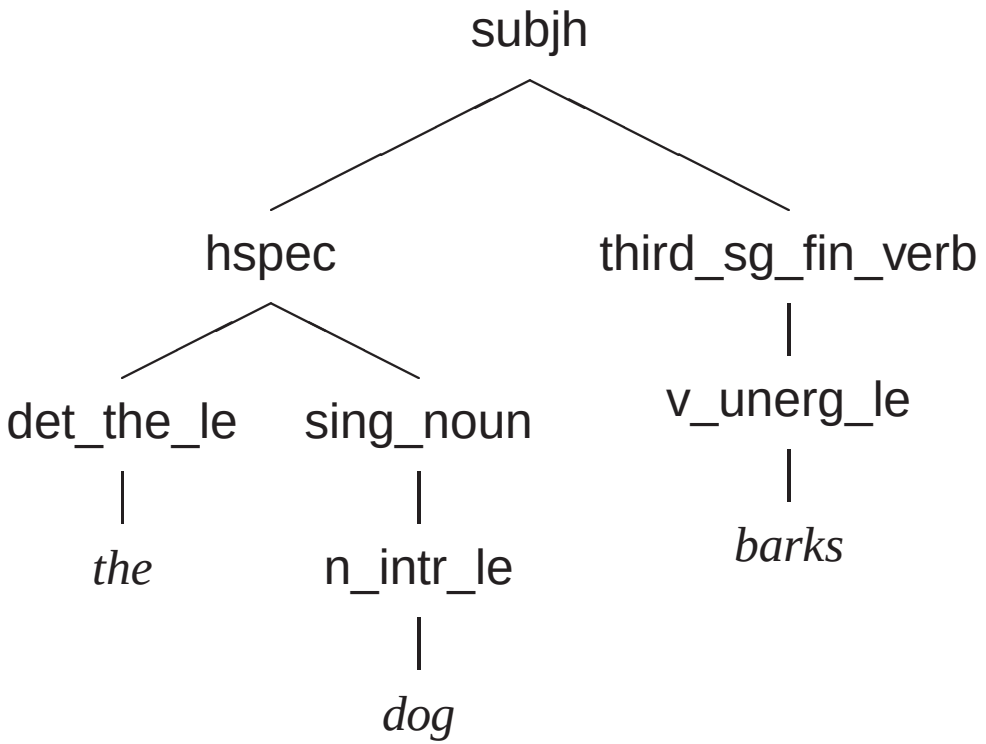
Aggregate	**items** ♯	**words** ϕ	**ambiguity** ϕ	**baseline** %
$100 \leq$ *readings*	87	20.5	580.8	0.42
$50 \leq$ *readings* < 100	61	17.3	73.0	1.44
$10 \leq$ *readings* < 50	269	15.1	22.5	5.61
$5 <$ *readings* < 10	172	11.1	6.9	15.66
$1 <$ *readings* < 5	275	8.8	2.8	40.9
Total	**864**	**13.0**	**72.9**	**18.03**

The treebank data binned with respect to generator ambiguity, for each group showing the total number of items, average string length, average number of paraphrases, and a random choice baseline for accuracy.

Maximum Entropy Models

- Given by a set of *features* $\{f_1, \ldots, f_m\}$ and a set of associated *weights* $\{\lambda_1, \ldots, \lambda_m\}$.
- The real-valued feature-functions describe relevant properties of the data items.
- The lambda weights determine the contribution or importance of each feature.
- Prob. of a realization r given a semantics s:
  $p(r|s) = \frac{1}{Z(s)} \exp\left(\sum_i \lambda_i f_i(r)\right)$
- *Learning* amounts to finding the optimal weights λ that maximize the likelihood of the training corpus.

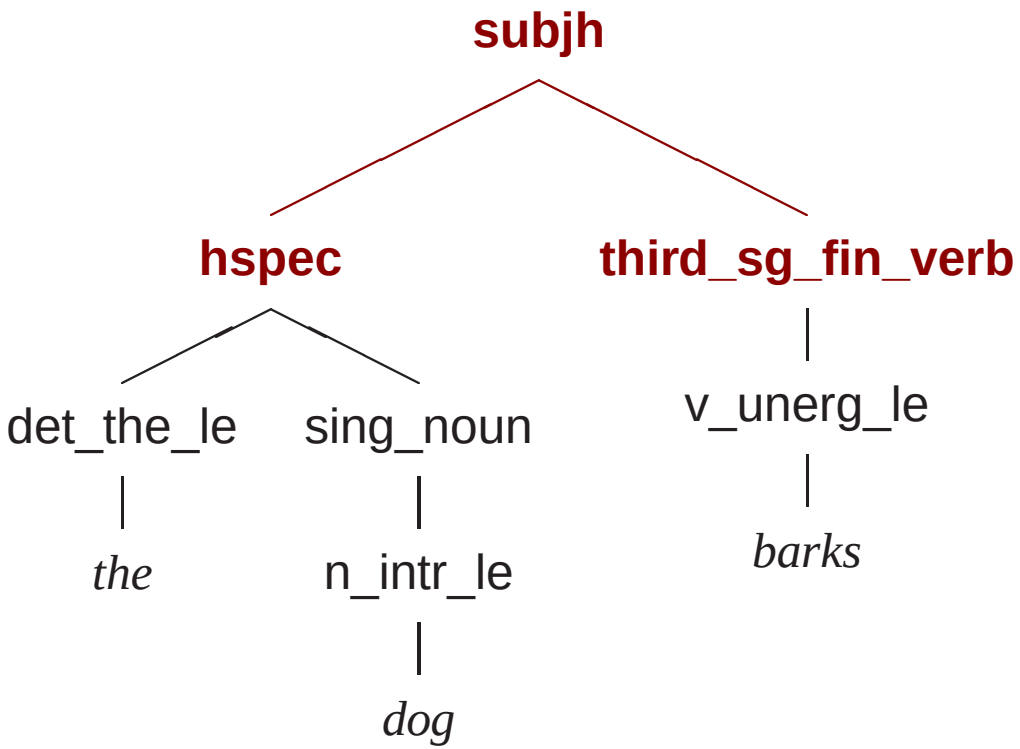
MaxEnt Features



Sample HPSG derivation tree for *the dog barks*.

Features record local derivation sub-trees with different degrees of lexicalization, levels of grandparenting, etc. Additional features record n-grams over lexical types.

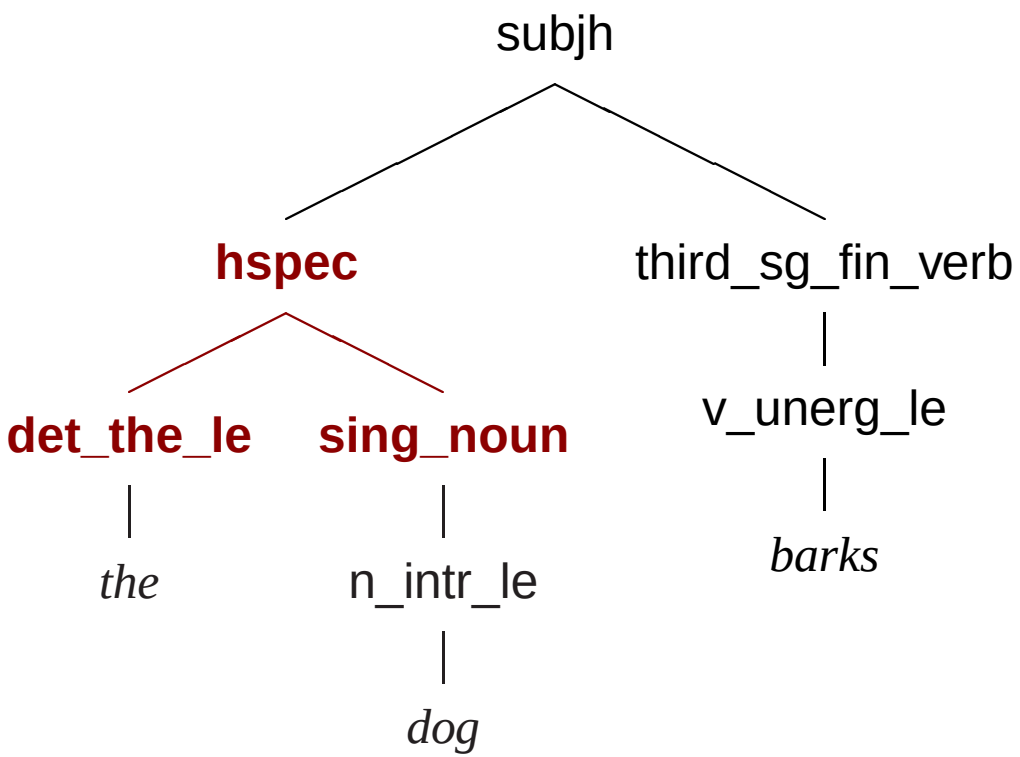
MaxEnt Features



Sample HPSG derivation tree for *the dog barks*.

Features record local derivation sub-trees with different degrees of lexicalization, levels of grandparenting, etc. Additional features record n-grams over lexical types.

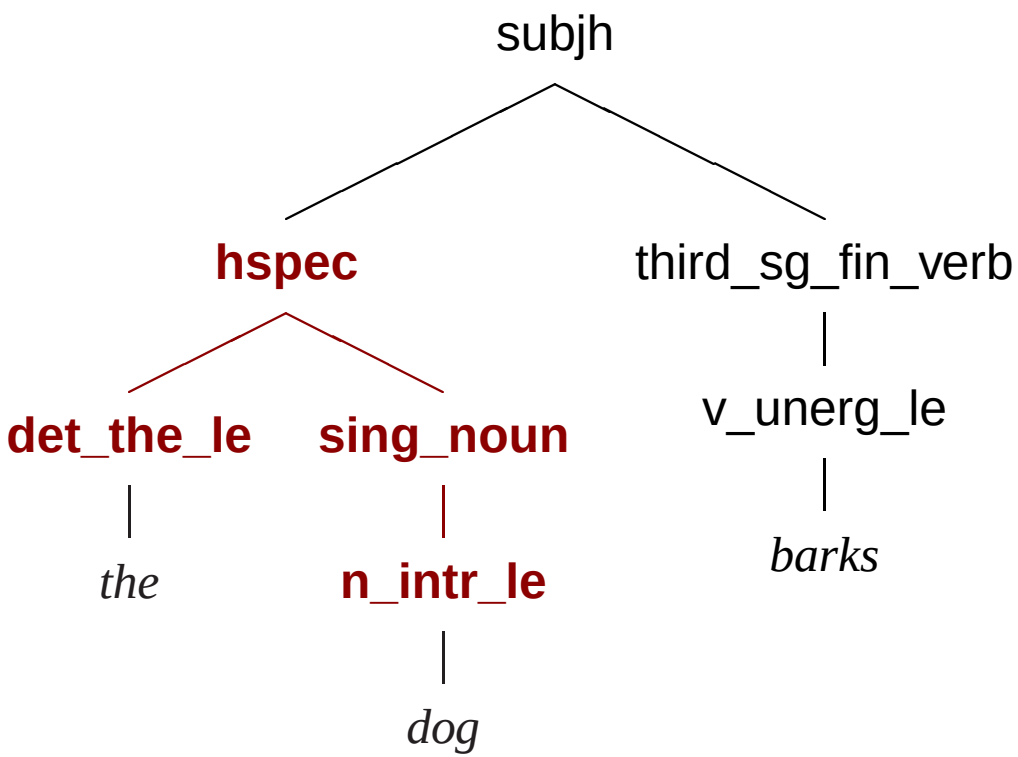
MaxEnt Features



Sample HPSG derivation tree for *the dog barks*.

Features record local derivation sub-trees with different degrees of lexicalization, levels of grandparenting, etc. Additional features record n-grams over lexical types.

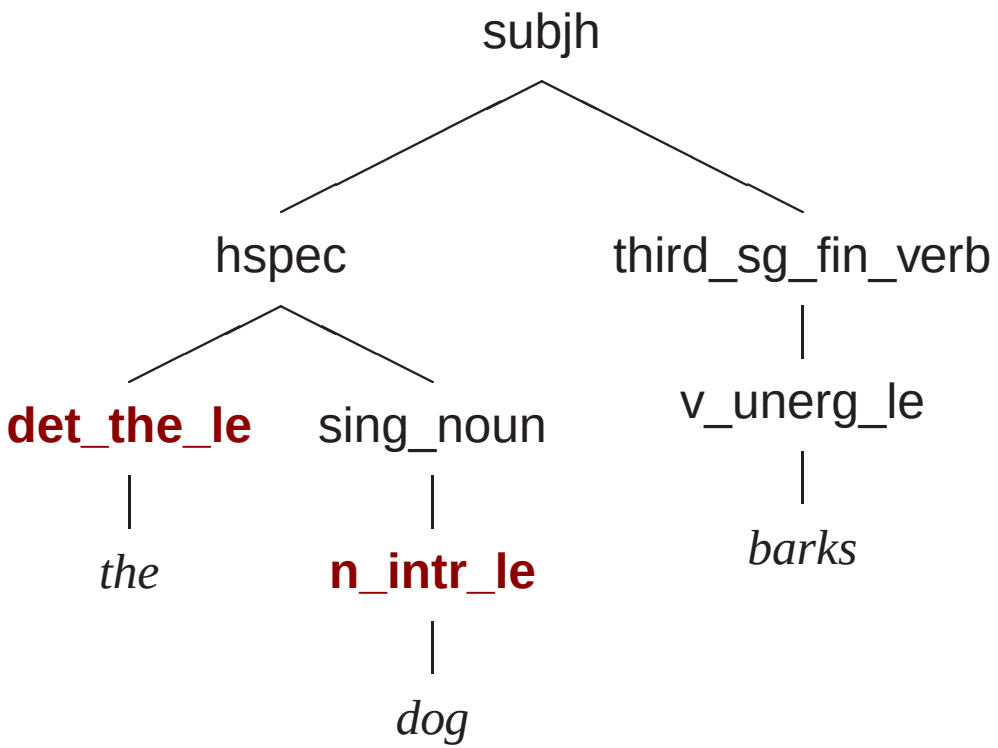
MaxEnt Features



Sample HPSG derivation tree for *the dog barks*.

Features record local derivation sub-trees with different degrees of lexicalization, levels of grandparenting, etc. Additional features record n-grams over lexical types.

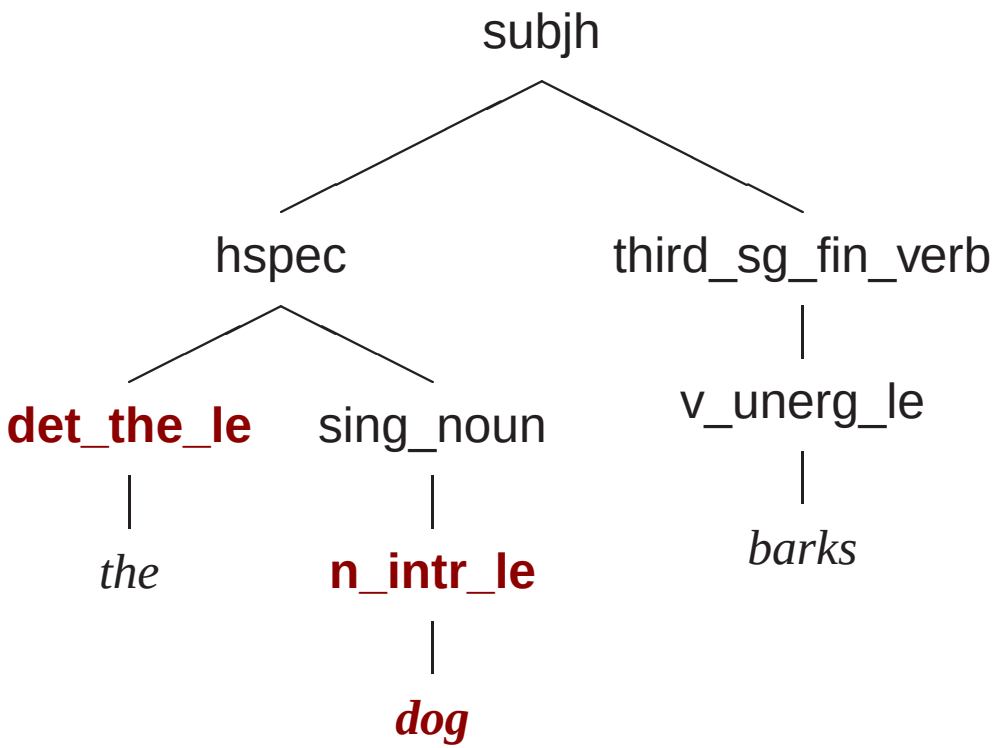
MaxEnt Features



Sample HPSG derivation tree for *the dog barks*.

Features record local derivation sub-trees with different degrees of lexicalization, levels of grandparenting, etc. Additional features record n-grams over lexical types.

MaxEnt Features



Sample HPSG derivation tree for *the dog barks*.

Features record local derivation sub-trees with different degrees of lexicalization, levels of grandparenting, etc. Additional features record n-grams over lexical types.

The MaxEnt Ranker

- Exact match accuracy: 61.58%
- BLEU: 0.903
- When training and testing by 10-fold cross-validation on the small 'Rondane' data set, we get results competitive with a language model trained on the entire BNC.
  - Structural features are a good thing.
  - Having training data attuned to the domain is a good thing.

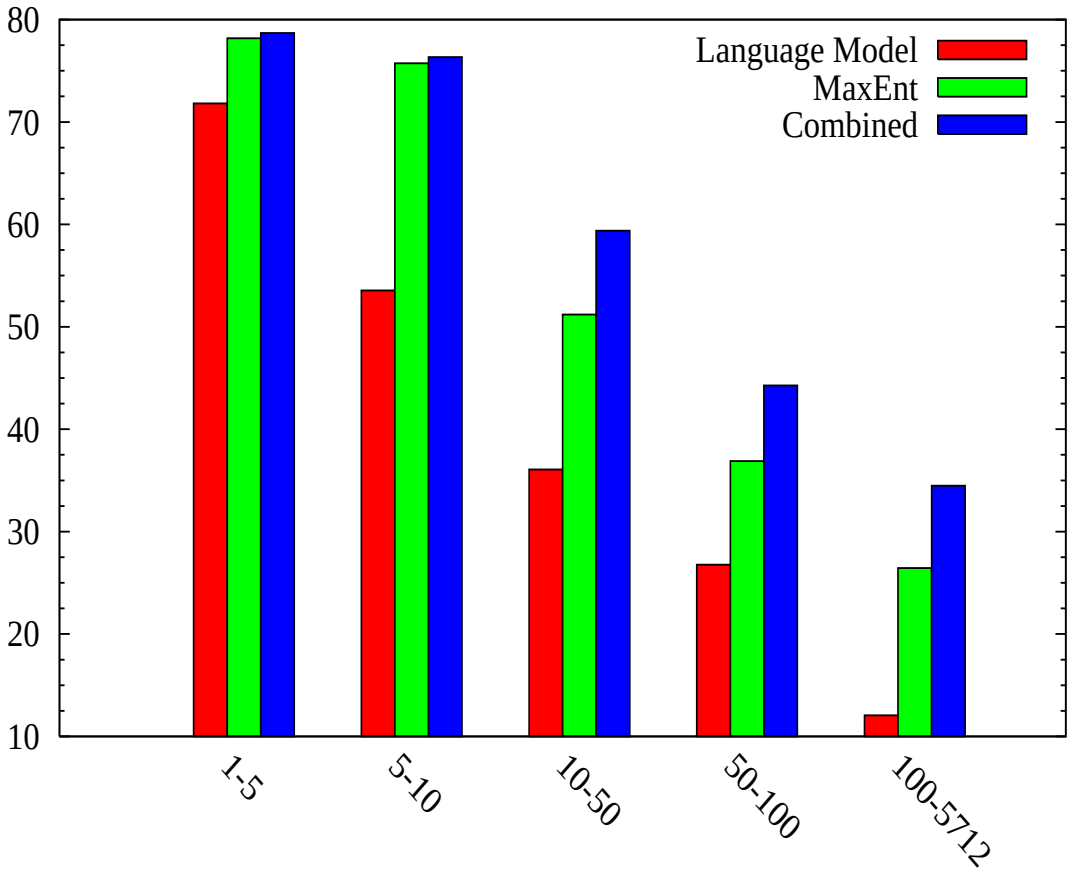
A Combined Ranker

- Many non-overlapping errors made by the different models, leaving more to be gained by combining the two.
- We can throw in the n-gram probabilities as a separate feature in the MaxEnt model to get a combined model.

A Combined Ranker

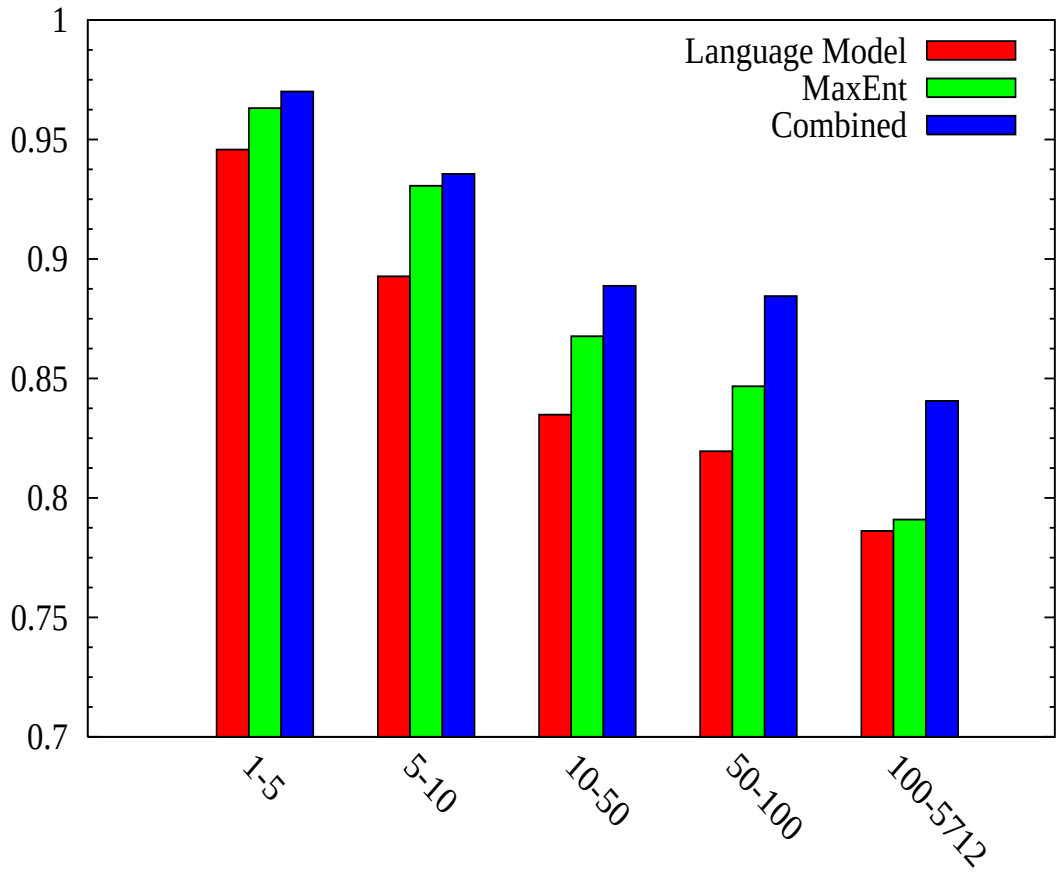
- Many non-overlapping errors made by the different models, leaving more to be gained by combining the two.
- We can throw in the n-gram probabilities as a separate feature in the MaxEnt model to get a combined model.
- Exact match accuracy: 65.63%
- BLEU: 0.920

Exact Match Accuracy



Exact match accuracy scores for the different models. Data items are binned with respect to number of distinct realizations.

BLEU



Averaged sentence-level BLEU scores for the different models. Data items are binned with respect to number of distinct realizations.

Summary

- Successful combination of linguistic grammar and stochastic disambiguation for target language generation in a hybrid MT system (LOGON).
- The ranking module benefits from combining statistics from different sources; surface oriented n-grams in addition to structural features of derivation trees.
- Ongoing work: Generation from packed MRSs and selective unpacking (Carroll & Oepen, 2005).

References

Callison-Burch, C., & Flournoy, R. S. (2001). A program for automatically selecting the best output from multiple machine translation engines. In *Proceedings of the MT Summit.* Santiago, Spain.

Carroll, J., Copestake, A., Flickinger, D., & Poznanski, V. (1999). An efficient chart generator for (semi-)-lexicalist grammars. In *Proceedings of the 7th European Workshop on Natural Language Generation* (pp. 86 – 95). Toulouse, France.

Carroll, J., & Oepen, S. (2005). High efficiency realization for a wide-coverage unification grammar. In R. D. and (Ed.), *Proceedings of the 2nd International Joint Conference on Natural Language Processing.* Jeju, Republic of Korea.

Copestake, A., Flickinger, D., Malouf, R., Riehemann, S., & Sag, I. (1995). Translation using minimal recursion semantics. In *Proceedings of the Sixth International Conference on Theoretical and Methodological Issues in Machine Translation.* Leuven, Belgium.

Flickinger, D. (2002). On building a more efficient grammar by exploiting types. In S. Oepen, D. Flickinger, J. Tsujii, , & H. Uszkoreit (Eds.), *Collaborative language engineering: A case study in efficient grammar-based processing* (pp. 1–17). CSLI Press.

Langkilde, I., & Knight, K. (1998). The practical value of n-grams in generation. In *International natural language generation workshop.*

Toutanova, K., Manning, C. D., Shieber, S. M., Flickinger, D., & Oepen, S. (2002). Parse disambiguation for a rich hpsg grammar. In *First workshop on treebanks and linguistic theories*. Sozopol, Bulgaria.

Velldal, E., Oepen, S., & Flickinger, D. (2004). Paraphrasing treebanks for stochastic realization ranking. In *Proceedings of the 3rd Workshop on Treebanks and Linguistic Theories*. Tübingen, Germany.

White, M. (2004). Reining in CCG chart realization. In *Proceedings of the 3rd International Conference on Natural Language Generation*. Hampshire, UK.