Computational linguistics and linguistics

Ann Copestake

Computer Laboratory
University of Cambridge

March 2007

What computational linguists do.

- Build applications: e.g., machine translation, 'intelligent' search, information extraction, summarization, email response, spoken dialogue systems.
- Investigate language processing 'modules' (possibly useful in applications): e.g., tokenisation, morphology, parsing, generation, word sense disambiguation, anaphora resolution, discourse segmentation, robust entailment.
- Develop resources for modules: e.g., lexical acquisition, grammar acquisition.
- Human language modelling (often with psycholinguists): e.g., language acquisition, productivity, semantic judgements.

Outline of this talk.

1. Linguistically-motivated computational grammars.
   - Overview.
   - DELPH-IN ERG and Matrix
   - Ambiguity, parse ranking and robustness.
2. 'Empirical' methods in computational linguistics.
   - An outline of corpus-based methodologies.
   - Learning placement of articles.
   - Collocation.

This is a personal perspective!

Outline.

1. Linguistically-motivated computational grammars.
   - Overview.
   - DELPH-IN ERG and Matrix
   - Ambiguity, parse ranking and robustness.
2. 'Empirical' methods in computational linguistics.
   - An outline of corpus-based methodologies.
   - Learning placement of articles.
   - Collocation.

Formalisms and systems

- Theoretical frameworks, generally hand-built grammars, partially hand-built lexicons, may have compositional semantics, may be bidirectional (parse and generate).
- LFG (PARC XLE system), TAG (U.Penn), CCG, GPSG, IBM/Microsoft, FUF/SURGE (generation only) ...
- HPSG: DELPH-IN (`http://www.delph-in.net`). Also TROLL, Tokyo system ...
- Last 10-15 years: stochastic parse ranking, automatic lexical acquisition, robustness, processing speed.
- Linguistically motivated grammars were dominant paradigm in 1980s: proportionally less significant now.

Outline.

1. Linguistically-motivated computational grammars.
   - Overview.
   - DELPH-IN ERG and Matrix
   - Ambiguity, parse ranking and robustness.
2. 'Empirical' methods in computational linguistics.
   - An outline of corpus-based methodologies.
   - Learning placement of articles.
   - Collocation.

The DELPH-IN English Resource Grammar (Flickinger et al)

- ERG demo at `http://erg.emmtee.net`.
- Broad-coverage, precise, bidirectional grammar for English, used in a number of projects.
- Approximately 80% coverage for corpora tried so far (after lexicon and any specific constructions added).
- Variety of strategies for adding lexicon automatically.
- Combine with 'shallow' analysers where robustness is required.
- Grammars for other languages developed partially on basis of the ERG (Japanese, German).

The DELPH-IN Grammar Matrix (Bender et al)

- A toolkit for grammar development: used for small grammars of lots of languages (teaching), more substantial grammars of Norwegian, Greek, Korean, Spanish, Swedish, Italian (research).
- Aim: provide a core grammar that can be specialised for individual languages.
- Some typological distinctions automatically converted into grammar fragments.
- Potential use for field linguists.

Coordination in the Matrix (Drellishak and Bender, 2005)

Four dimensions:

1. kind of marking (lexical, morphological, none).
2. pattern of Marking: a-, mono-, poly-, or omnisyndeton.
3. position of Marking: before or after the coordinand.
4. phrase types covered: NP, NOM, VP, AP, etc.

Consistent compositional semantics for all variants.

Coordination in the Matrix (Drellishak and Bender, 2005)

$$
\begin{bmatrix}
\textbf{coord-phrase} & \\
\text{SYNSEM.LOCAL.CAT} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{MOD} & \boxed{1} \\ \text{VAL} & \boxed{2} \end{bmatrix} \end{bmatrix} \\
\text{LCOORD-DTR} & \boxed{3} \begin{bmatrix} \textbf{sign} & \\ \text{SYNSEM.LOCAL.CAT} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{MOD} & \boxed{1} \\ \text{VAL} & \boxed{2} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\
\text{RCOORD-DTR} & \boxed{4} \begin{bmatrix} \textbf{sign} & \\ \text{SYNSEM.LOCAL.CAT} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{MOD} & \boxed{1} \\ \text{VAL} & \boxed{2} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\
\text{ARGS} & \langle \boxed{3}, \boxed{4} \rangle
\end{bmatrix}
$$

e.g., VP conjunction in Ono, inherits from **coord-phrase**:

$$
\begin{bmatrix}
\textbf{vp-top-coord-rule} & \\
\text{SYNSEM.LOCAL.CAT.HEAD.VFORM} & \boxed{5} \\
\text{LCOORD-DTR.SYNSEM.LOCAL.CAT.HEAD} & \begin{bmatrix} \text{VFORM} & \boxed{5} \\ \text{MEDIAL} & + \end{bmatrix} \\
\text{RCOORD-DTR.SYNSEM.LOCAL.CAT.HEAD} & \begin{bmatrix} \text{VFORM} & \boxed{5} \\ \text{MEDIAL} & - \end{bmatrix}
\end{bmatrix}
$$

Outline.

1. Linguistically-motivated computational grammars.
   - Overview.
   - DELPH-IN ERG and Matrix
   - Ambiguity, parse ranking and robustness.
2. 'Empirical' methods in computational linguistics.
   - An outline of corpus-based methodologies.
   - Learning placement of articles.
   - Collocation.

Ambiguity.

- Pre-1990s, most discussion related to lexical ambiguity, PP-attachment, etc.
  I saw the man with the telescope.
  Assumption was that real world knowledge was needed to resolve ambiguity.
- Early discussion of statistical techniques often suggested corpus-based information was an approximation for AI and inference.
- But for actual large coverage grammars, even high precision ones, most ambiguity is due to unusual constructions.
  Police save a lobster from certain death.
  save as preposition, imperative etc.

Ambiguity.

- Pre-1990s, most discussion related to lexical ambiguity, PP-attachment, etc.
  I saw the man with the telescope.
  Assumption was that real world knowledge was needed to resolve ambiguity.
- Early discussion of statistical techniques often suggested corpus-based information was an approximation for AI and inference.
- But for actual large coverage grammars, even high precision ones, most ambiguity is due to unusual constructions.
  Police save a lobster from certain death.
  save as preposition, imperative etc.

Ambiguity.

- Pre-1990s, most discussion related to lexical ambiguity, PP-attachment, etc.
  I saw the man with the telescope.
  Assumption was that real world knowledge was needed to resolve ambiguity.
- Early discussion of statistical techniques often suggested corpus-based information was an approximation for AI and inference.
- But for actual large coverage grammars, even high precision ones, most ambiguity is due to unusual constructions.
  Police save a lobster from certain death.
  save as preposition, imperative etc.

Parse ranking.

- Build a treebank by automatically parsing sentences in a corpus and manually disambiguating (or, Penn Treebank, manually construct trees with aid of a robust grammar).
- Train parse ranking algorithm on the saved trees.
- ERG etc: Redwoods Treebank. Note need to update treebank for new versions of grammar.
- Redwoods experience: around 3000 sentences gives good model: many more would be needed for lexical dependencies (e.g., *saw/telescope* vs *man/telescope*).

Lexical probabilities.

- Word sense frequencies, subcategorisation possibilities etc generally have very skewed distributions.
- e.g., *diet* (food) is 100 times more frequent than *diet* (parliament) in BNC.
- e.g., *believe* `NP VPinf` very infrequent compared to *believe* `NP`, *believe* `that S`.
- Similar effect for subcat frames related by alternation.
- Variation over time, etc (e.g., Briscoe, 2001) causes problems for normal statistical techniques.

Grammaticality and subcategorization (Manning 2003).

Judgements (from Pollard and Sag, 1994):

1. We consider Kim an acceptable candidate
2. We consider Kim quite acceptable
3. *We consider Kim as an acceptable candidate
4. *We consider Kim as quite acceptable

Corpus data:

1. The boys consider her as family and she participates in everything we do.
2. 'We consider that as part of the job', Keep said.
3. ...he said he considers them as having championship potential.

Outline.

1. Linguistically-motivated computational grammars.
   - Overview.
   - DELPH-IN ERG and Matrix
   - Ambiguity, parse ranking and robustness.
2. 'Empirical' methods in computational linguistics.
   - An outline of corpus-based methodologies.
   - Learning placement of articles.
   - Collocation.

How to write an ACL paper, option 1.

1. Think of a phenomenon to work on: e.g., non-referential *it*.
2. Obtain a corpus, develop an annotation scheme: e.g., referential vs extraposed vs . . .
3. Mark up some part of a corpus (or pay someone else to do it).
4. Calculate agreement between annotators.
5. Find agreement is low: either a) return to 2 (good practice) or b) collapse categories until it's OK (bad practice).
6. Think of some features, train a machine learning algorithm (or several, since that's just as easy).
7. Report results which are better than baseline (a basic method) but lower than human agreement.

How to write an ACL paper, option 2.

1. Take someone else's marked up corpus.
2. Try a new machine learning method.
3. Report results which are better than the previous ones.

Variants: learn categories, lexicon etc.

Playing the annotation game.

- Generally, little knowledge of computer science or programming needed: scripting language to extract features (e.g., perl), standard machine learning packages (e.g., WEKA).
- Choice of phenomenon, development of annotation scheme, feature selection, error analysis: should all be informed by linguistics, currently often not done well.
- What does and doesn't work may be theoretically interesting.

Outline.

1. Linguistically-motivated computational grammars.
   - Overview.
   - DELPH-IN ERG and Matrix
   - Ambiguity, parse ranking and robustness.

2. 'Empirical' methods in computational linguistics.
   - An outline of corpus-based methodologies.
   - Learning placement of articles.
   - Collocation.

Machine learning of articles.

Task: take an English corpus with noun phrases marked, remove all instances of *the* and *a*, predict *the* vs *a* vs no determiner.

```
(S
 (NP-SBJ
  (NP no (NNP Pierre) (NNP Vinken) )
  (, ,)
  (ADJP
   (NP no (CD 61) (NNS years) )
   (JJ old) )
  (, ,) )
 (VP (MD will)
  (VP (VB join)
   (NP (DT the) (NN board) )
   (PP-CLR (IN as)
    (NP (DT a) (JJ nonexecutive) (NN director) ))
   (NP-TMP no (NNP Nov.) (CD 29) )))
 (. .) )
```

Results on WSJ: baseline (no article): 70%, head of NP: 80%, combined features: 83%.
No improvement with 'discourse' features.

Machine learning of articles.

Task: take an English corpus with noun phrases marked, remove all instances of *the* and *a*, predict *the* vs *a* vs no determiner.

```
(S
 (NP-SBJ
  (NP no (NNP Pierre) (NNP Vinken) )
  (, ,)
  (ADJP
   (NP no (CD 61) (NNS years) )
   (JJ old) )
  (, ,) )
 (VP (MD will)
  (VP (VB join)
   (NP (DT the) (NN board) )
   (PP-CLR (IN as)
    (NP (DT a) (JJ nonexecutive) (NN director) ))
   (NP-TMP no (NNP Nov.) (CD 29) )))
 (. .) )
```

Results on WSJ: baseline (no article): 70%, head of NP: 80%, combined features: 83%.

No improvement with 'discourse' features.

Machine learning of articles.

Task: take an English corpus with noun phrases marked, remove all instances of *the* and *a*, predict *the* vs *a* vs no determiner.

```
(S
 (NP-SBJ
  (NP no (NNP Pierre) (NNP Vinken) )
  (, ,)
  (ADJP
   (NP no (CD 61) (NNS years) )
   (JJ old) )
  (, ,) )
 (VP (MD will)
  (VP (VB join)
   (NP (DT the) (NN board) )
   (PP-CLR (IN as)
    (NP (DT a) (JJ nonexecutive) (NN director) ))
   (NP-TMP no (NNP Nov.) (CD 29) )))
 (. .) )
```

Results on WSJ: baseline (no article): 70%, head of NP: 80%, combined features: 83%.

No improvement with 'discourse' features.

Outline.

Collocation.

BNC frequencies:

	number	proportion	quality	problem	part	winds	rain
large	1790	404	0	10	533	0	0
high	92	501	799	0	3	90	0
big	11	1	0	79	79	3	1
heavy	0	0	1	0	1	2	198

Informal acceptability judgements:

	number	proportion	quality	problem	part	winds	rain
large			*			*	*
high				*	?		*
big			?				*
heavy	?	*	*	*			

Magnitude adjective distribution.

- Investigated the distribution of *heavy, high, big, large, strong, great, major* with the most common co-occurring nouns in the BNC.
- Nouns tend to occur with up to three of these adjectives with high frequency and low or zero frequency with the rest.
- 50 nouns in BNC with the extended use of *heavy* with frequency 10 or more, 160 such nouns with *high*. Only 9 with both: *price, pressure, investment, demand, rainfall, cost, costs, concentration, taxation*
- Clusters: e.g., weather precipitation nouns with *heavy*. Note *heavy shower* (weather, not bathroom).

Hypotheses about distribution.

- 'abstract' *heavy, high, big, large, strong, great, major* all denote magnitude (in a way that can be made formally precise)
- distribution differences due to collocation, soft rather than hard constraints
- adjective-noun combination is semi-productive
- denotation and syntax allow *heavy esteem* etc, but speakers are sensitive to frequencies, prefer more frequent phrases with same meaning

Two quotations.

All continuities, all possibilities of infinitesimal gradation, are shoved outside of linguistics in one direction or the other. Joos (1950)

It must be recognized that the notion 'probability of a sentence' is an entirely useless one, under any known interpretation of this term. (Chomsky 1969)

If you believe this, then you won't like modern computational linguistics!

Two quotations.

> *All continuities, all possibilities of infinitesimal gradation, are shoved outside of linguistics in one direction or the other. Joos (1950)*

> *It must be recognized that the notion 'probability of a sentence' is an entirely useless one, under any known interpretation of this term. (Chomsky 1969)*

If you believe this, then you won't like modern computational linguistics!

Conclusion.

- Computational linguists work with linguists on 'traditional' models including morphology, syntax and compositional semantics.
- Computational linguistics has demonstrated that probabilistic models can provide better models of language than purely symbolic ones.
  - This is not simply due to performance, world knowledge, pragmatics or other non-linguistically relevant effects.
  - If 'linguistics' means the study of language, then this is part of linguistics . . .
- Huge amounts of work remain to be done at a theoretical and methodological level.