Dependency Grammar and Dependency Parsing

Joakim Nivre

Växjö University and Uppsala University

Warning

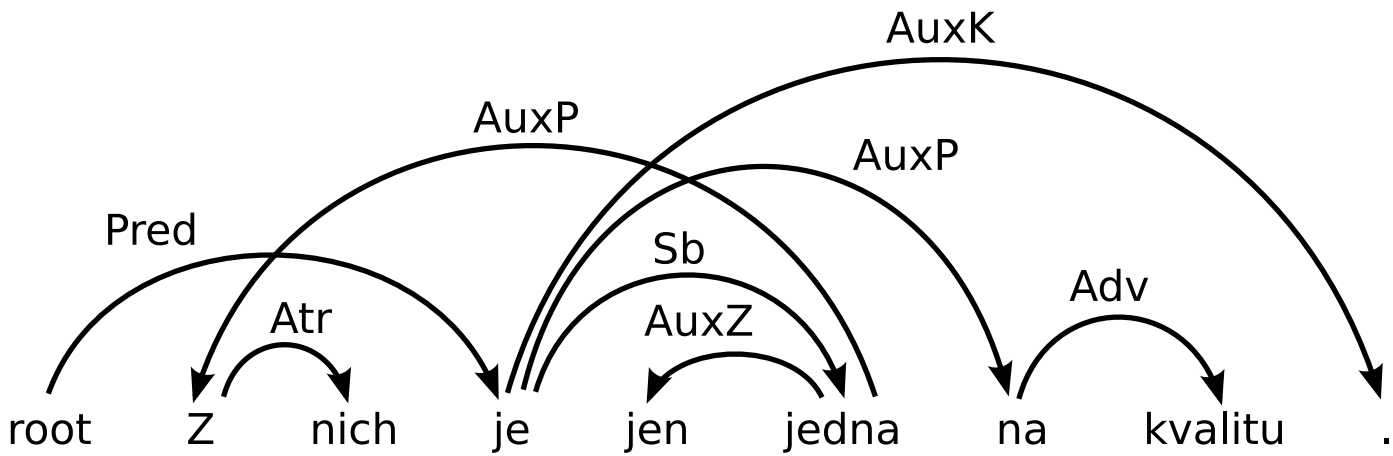
- This talk is different from some of the previous talks, because
- dependency grammar is not a unified framework, and
- many dependency parsers don't use grammars at all.

Outline

- Dependency grammar:
  - What it is and what it isn't
- Dependency parsing:
  - What, why and how?
- Dependency treebanks:
  - Proper and converted treebanks
- Multilingual dependency parsing:
  - The CoNLL shared tasks 2006 and 2007
- Conclusion and outlook

Dependency Grammar

- The basic idea:
  - Syntactic structure consists of lexical items, linked by binary asymmetric relations called dependencies.



("Only one of them concerns quality.")

Dependency Structure and Phrase Structure

- Dependency structures explicitly represent
  - head-dependent relations (directed arcs),
  - functional categories (arc labels),
  - possibly some structural categories (parts-of-speech).
- Phrase structures explicitly represent
  - phrases (nonterminal nodes),
  - structural categories (nonterminal labels),
  - possibly some functional categories (grammatical functions).
- Note:
  - Hybrid representations may combine all elements.
  - Discontinuous structures may be (dis)allowed in both.

Some Theoretical Frameworks

- Word Grammar (WG) [Hudson 1984, Hudson 1990]
- Functional Generative Description (FGD) [Sgall et al. 1986]
- Dependency Unification Grammar (DUG)
  [Hellwig 1986, Hellwig 2003]
- Meaning-Text Theory (MTT) [Mel'čuk 1988]
- (Weighted) Constraint Dependency Grammar ([W]CDG)
  [Maruyama 1990, Harper and Helzerman 1995,
  Menzel and Schröder 1998, Schröder 2002]
- Functional Dependency Grammar (FDG)
  [Tapanainen and Järvinen 1997, Järvinen and Tapanainen 1998]
- Topological/Extensible Dependency Grammar ([T/X]DG)
  [Duchier and Debusmann 2001, Debusmann et al. 2004]

Some Theoretical Issues

- Dependency structure sufficient as well as necessary?
- Mono-stratal or multi-stratal syntactic representations?
- What is the nature of lexical elements (nodes)?
  - Morphemes?
  - Word forms?
  - Multi-word units?
- What is the nature of dependency types (arc labels)?
  - Grammatical functions?
  - Semantic roles?
- What are the criteria for identifying heads and dependents?
- What are the formal properties of dependency structures?

Conclusion – Dependency Grammar

- The theoretical tradition of dependency grammar is united by
  - the assumption that syntactic structure resides in binary asymmetrical relations between lexical elements,
  - a common analysis of core syntactic constructions (predicate-argument and head-modifier structures).
- However, there are also important differences with respect to
  - whether dependency analysis exhausts syntactic analysis,
  - what the exact formal properties of dependency structures are,
  - how certain syntactic constructions should be analyzed.

Dependency Parsing

- Dependency parsing in a wide sense:
  - Any approach to parsing that makes use of word-to-word dependencies. For example:
    - Lexicalized statistical parsers [Collins 1999, Charniak 2000]
    - Parsers based on lexicalist grammar formalisms (LFG, HPSG, CCG, LTAG, ...)
- Dependency parsing in a narrow sense:
  - Any approach to parsing where the output is a dependency-based representation. In particular:
    - Mapping a sentence to a single (labeled) dependency graph.

Why Dependency Parsing?

- Recent surge of interest in dependency parsing
- Apparently because dependency-based representations
  - have a natural way of representing discontinuous constructions, making them suitable for free word order languages,
  - appear to be useful in many NLP applications, because of a transparent encoding of predicate-argument structure,
  - can be derived very efficiently, because of limited expressivity and complexity.

Major Approaches

- Grammar-based parsing:
  - Context-free dependency grammar [Hays 1964, Gaifman 1965]:
    - Standard chart parsing techniques
    - Modern descendant: Alpino (Dutch) [Bouma et al. 2000]
  - Constraint dependency grammar [Maruyama 1990]:
    - Parsing as constraint satisfaction
    - Modern descendant: WCDG (German) [Foth et al. 2004]
- Data-driven parsing:
  - Graph-based models [Eisner 1996]:
    - Learn to score dependency graphs factored by their arcs
    - Example: MSTParser [McDonald and Pereira 2006]
  - Transition-based models [Kudo and Matsumoto 2002]:
    - Learn to predict the next parser action given the parse history
    - Example: MaltParser [Nivre et al. 2007b]

The Role of Annotation

- Syntactically annotated corpora, or treebanks, are
  - essential for training and tuning of data-driven parsers,
  - useful in the development of grammar-based parsers, e.g., for grammar induction and parse selection,
  - necessary for the evaluation of all kinds of parsers.
- For dependency parsing, we need dependency treebanks, containing sentences annotated with dependency graphs.

Dependency Treebanks

- Dependency treebanks proper:
  - Sentences annotated with dependency structures
  - No conversion needed
  - Example: Prague Dependency Treebank (Czech)
- Converted treebanks:
  - Sentences annotated with other representations
  - Accuracy of conversion depends on properties of the annotation
  - Examples:
    - Alpino Treebank (Dutch): Hybrid annotation including both phrase and dependency structure; conversion limited to (deterministic) extraction
    - Penn Treebank (English): Phrase structure with sparse grammatical functions; conversion dependent on (more or less precise) heuristics

The PDT Model

- Prague Dependency Treebank [Hajič et al. 2001]:
  - 1.5 million words
  - Annotation in three layers:
    - Morphological annotation
    - Analytical annotation (surface syntax)
    - Tectogrammatical annotation (deep syntax)
  - Used extensively for parsing experiments
- Followers (analytical layer):
  - Prague Arabic Dependency Treebank
  - Slovene Dependency Treebank
  - Greek Dependency Treebank

Treebank Conversion

- Structural conversion:
  - Converting phrase structure to dependency structure requires finding exactly one head child in every phrase.
  - If this information is not part of the original annotation, heuristic head finding rules have to be used.
- Labels:
  - Labeling dependencies requires that every non-head child of a phrase is assigned a dependency label.
  - If this information is not part of the original annotation, heuristic labeling rules have to be used.
- Non-local dependencies:
  - Empty categories may need to be replaced by their antecedents to capture non-local dependencies.

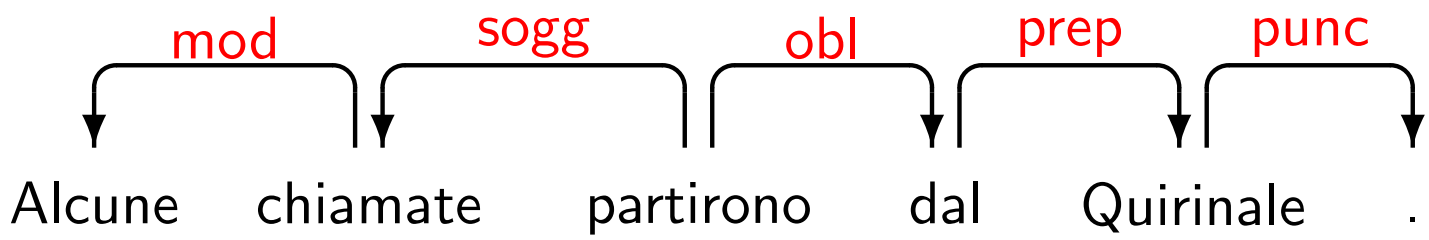
The CoNLL Shared Tasks

- CoNLL shared task 2006 and 2007:
  - Multilingual dependency parsing
  - Train a single parser on data from multiple languages
  - Parsers evaluated by labeled attachment score (LAS):
    - Percentage of tokens with correct head and label
- Scope and participation:
  - 2006: 13 languages, 19 submissions [Buchholz and Marsi 2006]
  - 2007: 10 languages, 23 submissions [Nivre et al. 2007a]

Data Format

- Text-based format with 8(10) tab-separated columns:
  - One line per token
  - Sentences separated by blank lines

INPUT						OUTPUT	
ID	**FORM**	**LEMMA**	**CPOSTAG**	**POSTAG**	**FEATS**	**HEAD**	**DEPREL**
1	Alcune	alcuna	D	DI	gen=F|num=P	2	mod
2	chiamate	chiamata	S	S	gen=F|num=P	3	sogg
3	partirono	partire	V	V	num=P|per=3|mod=I|tmp=R	0	ROOT
4	dal	da	E	E	gen=M|num=S	3	obl
5	Quirinale	quirinale	S	SP	gen=N|num=N	4	prep
6	.	.	PU	PU	_	5	punc



"Some calls originated from Quirinale."

Languages and Treebanks

- Languages:
  - Arabic*, Basque*, Bulgarian, Catalan, Chinese, Czech*, Danish*, Dutch, English, German, Greek*, Hungarian, Italian, Japanese, Portuguese, Slovene*, Spanish, Swedish, Turkish*
- Treebanks:
  - About one third genuine dependency treebanks
  - About two thirds converted treebanks (varying accuracy)
  - Number of dependency labels ranging from 7 to 82
  - Training set sizes ranging from 30k to 1.2M tokens
- Data sets available and reusable for future research

Results

- Best performing systems:
  - Graph-based approaches
  - Transition-based approaches
  - Ensemble systems (in 2007)
- Huge variation in parsing accuracy across languages:
  - Highest top score: 91.7% (Japanese 2006)
  - Lowest top score: 65.7% (Turkish 2006)
- Factors explaining variation:
  - Language type
  - Annotation scheme
  - Training set size
  - Sentence complexity

NB: Hard to control explanatory variables

Conclusion

- Tremendous achievements in recent years, culminating with the CoNLL shared tasks:
  - Treebanks with dependency annotation (possibly converted) available for some twenty languages
  - Dependency parsers (in the narrow sense) available for at least as many languages
- Concerns for the future:
  - Many of the treebanks are conversions of unknown quality
  - Many of the treebanks are small
  - Parsers only deal with surface syntax (and not even all of that)
  - Lack of comparability across languages and treebanks
  - No unified theoretical framework for annotation or parsing

Challenges

- Unified linguistic annotation?
  - Across languages:
    - Why is parsing accuracy 80% for Czech and 90% for English?
    - Richly inflected language vs. configurational language?
    - Dependency annotation vs. conversion from phrase structure?
    - A common annotation framework increases comparability.
  - Across linguistic layers:
    - How good is 90% labeled parsing accuracy?
    - Good enough to support semantic interpretation?
    - Unified syntactic-semantic annotation facilitates evaluation.

The CoNLL Shared Task 2008

- Joint inference of syntactic and semantic dependencies:
  - Semantic role labeling on a dependency-based representation
  - Integrated syntactic and semantic annotation based on:
    - Penn Treebank [Marcus et al. 1993] (enhanced conversion)
    - PropBank [Palmer et al. 2005] (verbal predicates)
    - NomBank [Meyers et al. 2005] (nominal predicates)
- Shared task website:

  http://www.yr-bcn.es/conll2008

- Gosse Bouma, Gertjan van Noord, and Robert Malouf. 2000. Alpino: Wide-coverage computational analysis of dutch. In *Computational Linguistics in the Netherlands 2000. Selected Papers from the Eleventh CLIN Meeting*, pages 45–59. Rodopi.
- Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL)*, pages 149–164.
- Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 132–139.
- Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.
- Ralph Debusmann, Denys Duchier, and Geert-Jan M. Kruijff. 2004. Extensible dependency grammar: A new methodology. In *Proceedings of the Workshop on Recent Advances in Dependency Grammar*, pages 78–85.
- Denys Duchier and Ralph Debusmann. 2001. Topological dependency trees: A constraint-based account of linear precedence. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 180–187.
- Jason M. Eisner. 1996. Three new probabilistic models for dependency parsing: An exploration. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, pages 340–345.
- Kilian Foth, Michael Daum, and Wolfgang Menzel. 2004. A broad-coverage parser for German based on defeasible constraints. In *Proceedings of KONVENS 2004*, pages 45–52.
- Haim Gaifman. 1965. Dependency systems and phrase-structure systems. *Information and Control*, 8:304–337.
- Jan Hajič, Barbora Vidova Hladka, Jarmila Panevová, Eva Hajičová, Petr Sgall, and Petr Pajas. 2001. Prague Dependency Treebank 1.0. LDC, 2001T10.
- Mary P. Harper and Randall A. Helzerman. 1995. Extensions to constraint dependency parsing for spoken language processing. *Computer Speech and Language*, 9:187–234.
- David G. Hays. 1964. Dependency theory: A formalism and some observations. *Language*, 40:511–525.

- Peter Hellwig. 1986. Dependency unification grammar. In *Proceedings of the 11th International Conference on Computational Linguistics (COLING)*, pages 195–198.
- Peter Hellwig. 2003. Dependency unification grammar. In Vilmos Agel, Ludwig M. Eichinger, Hans-Werner Eroms, Peter Hellwig, Hans Jürgen Heringer, and Hening Lobin, editors, *Dependency and Valency*, pages 593–635. Walter de Gruyter.
- Richard A. Hudson. 1984. *Word Grammar*. Blackwell.
- Richard A. Hudson. 1990. *English Word Grammar*. Blackwell.
- Timo Järvinen and Pasi Tapanainen. 1998. Towards an implementable dependency grammar. In *Proceedings of the Workshop on Processing of Dependency-Based Grammars (ACL-COLING)*, pages 1–10.
- Taku Kudo and Yuji Matsumoto. 2002. Japanese dependency analysis using cascaded chunking. In *Proceedings of the Sixth Workshop on Computational Language Learning (CoNLL)*, pages 63–69.
- Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19:313–330.
- Hiroshi Maruyama. 1990. Structural disambiguation with constraint propagation. In *Proceedings of the 28th Meeting of the Association for Computational Linguistics (ACL)*, pages 31–38.
- Ryan McDonald and Fernando Pereira. 2006. Online learning of approximate dependency parsing algorithms. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 81–88.
- Igor Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press.
- Wolfgang Menzel and Ingo Schröder. 1998. Decision procedures for dependency parsing using graded constraints. In *Proceedings of the Workshop on Processing of Dependency-Based Grammars (ACL-COLING)*, pages 78–87.
- Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2005. The nombank project: An interim report. Technical Report.

- Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007a. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task of EMNLP-CoNLL 2007*, pages 915–932.
- Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007b. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13:95–135.
- Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank. *Computational Linguistics*, 31:71–106.
- Ingo Schröder. 2002. *Natural Language Parsing with Graded Constraints*. Ph.D. thesis, Hamburg University.
- Petr Sgall, Eva Hajičová, and Jarmila Panevová. 1986. *The Meaning of the Sentence in Its Pragmatic Aspects*. Reidel.
- Daniel Sleator and Davy Temperley. 1993. Parsing English with a link grammar. In *Proceedings of the Third International Workshop on Parsing Technologies (IWPT)*, pages 277–292.
- Pasi Tapanainen and Timo Järvinen. 1997. A non-projective dependency parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP)*, pages 64–71.