Dependency Grammar

- Not a coherent grammatical framework: wide range of different kinds of DG
  - just as there are wide ranges of "generative syntax"
- Different core ideas than phrase structure grammar
- We will base a lot of our discussion on [Mel’čuk(1988)]

Dependency grammar is important for those interested in CL:
- Increasing interest in dependency-based approaches to syntactic parsing in recent years (e.g., CoNLL-X shared task, 2006)
Dependency Syntax

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

- In the (translated) words of Lucien Tesnière [Tesnière(1959)]:
  - The sentence is an organized whole, the constituent elements of which are words. [1.2] Every word that belongs to a sentence ceases by itself to be isolated as in the dictionary. Between the word and its neighbors, the mind perceives connections, the totality of which forms the structure of the sentence. [1.3] The structural connections establish dependency relations between the words. Each connection in principle unites a superior term and an inferior term. [2.1] The superior term receives the name governor. The inferior term receives the name subordinate. Thus, in the sentence *Alfred parle* [. . . ], *parle* is the governor and *Alfred* the subordinate. [2.2]
Overview: constituency

(1) Small birds sing loud songs

What you might be more used to seeing:
Overview: dependency

The corresponding dependency tree representations [Hudson(2000)]:

- **Small birds sing loud songs**
  - sbj: **Small**
  - obj: **birds**
  - nmod: **sing**
  - nmod: **loud**
  - nmod: **songs**

- **sing**
  - sbj: **birds**
  - obj: **songs**
  - nmod: **small**
  - nmod: **loud**
Constituency vs. Relations

- DG is based on relationships between words, i.e., dependency relations
  - A → B means A governs B or B depends on A ...
  - Dependency relations can refer to syntactic properties, semantic properties, or a combination of the two
    - Some variants of DG separate syntactic and semantic relations by representing different layers of dependency structures
  - These relations are generally things like subject, object/complement, (pre-/post-)adjunct, etc.
    - Subject/Agent: John fished.
    - Object/Patient: Mary hit John.

- PSG is based on groupings, or constituents
  - Grammatical relations are not usually seen as primitives, but as being derived from structure
Simple relation example

For the sentence *John loves Mary*, we have the relations:

- \(\text{loves} \rightarrow_{\text{subj}} \text{John}\)
- \(\text{loves} \rightarrow_{\text{obj}} \text{Mary}\)

Both *John* and *Mary* depend on *loves*, which makes *loves* the head, or **root**, of the sentence (i.e., there is no word that governs *loves*)

- The structure of a sentence, then, consists of the set of pairwise relations among words.
Economic news had little effect on financial markets.
## Terminology

<table>
<thead>
<tr>
<th>Superior</th>
<th>Inferior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head</td>
<td>Dependent</td>
</tr>
<tr>
<td>Governor</td>
<td>Modifier</td>
</tr>
<tr>
<td>Regent</td>
<td>Subordinate</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>
Notational Variants

had news Economic nmod
effect little on pc
tiny on financial nmod
markets pc

Dependency Grammar 10(29)
Economic news had little effect on financial markets.
Economic news had little effect on financial markets.
Economic news had little effect on financial markets.
Phrase Structure

Economic news had little effect on financial markets.
Comparison

- 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).

- Hybrid representations may combine all elements.
Some Theoretical Frameworks

- **Word Grammar (WG)** [Hudson(1984), Hudson(1990)]
- **Functional Generative Description (FGD)**
  [Sgall et al.(1986)Sgall, Hajičová and Panevová]
- **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),
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?
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?
Criteria for Heads and Dependents

Criteria for a syntactic relation between a head $H$ and a dependent $D$ in a construction $C$ [Zwicky(1985), Hudson(1990)]:

1. $H$ determines the syntactic category of $C$; $H$ can replace $C$.
2. $H$ determines the semantic category of $C$; $D$ specifies $H$.
3. $H$ is obligatory; $D$ may be optional.
4. $H$ selects $D$ and determines whether $D$ is obligatory.
5. The form of $D$ depends on $H$ (agreement or government).
6. The linear position of $D$ is specified with reference to $H$.

Issues:

- Syntactic (and morphological) versus semantic criteria
- Exocentric versus endocentric constructions
## Some Clear Cases

<table>
<thead>
<tr>
<th>Construction</th>
<th>Head</th>
<th>Dependent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exocentric</td>
<td>Verb</td>
<td>Subject (\textit{sbj})</td>
</tr>
<tr>
<td></td>
<td>Verb</td>
<td>Object (\textit{obj})</td>
</tr>
<tr>
<td>Endocentric</td>
<td>Verb</td>
<td>Adverbial (\textit{vmod})</td>
</tr>
<tr>
<td></td>
<td>Noun</td>
<td>Attribute (\textit{nmod})</td>
</tr>
</tbody>
</table>

Economic news suddenly affected financial markets.
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

I can see that they rely on this and that.
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

I can see that they rely on this and that.
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

I can see that they rely on this and that.

dependency analysis diagram

I  can  see  that  they  rely  on  this  and  that .
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

I can see that they rely on this and that.
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

I can see that they rely on this and that.
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

```dependency
I can see that they rely on this and that.
```
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

I can see that they rely on this and that.

```
I can see that they rely on this and that.
```

Dependency Grammar 17(29)
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

I can see that they rely on this and that.
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation
Some Tricky Cases

- Complex verb groups (auxiliary ↔ main verb)
- Subordinate clauses (complementizer ↔ verb)
- Coordination (coordinator ↔ conjuncts)
- Prepositional phrases (preposition ↔ nominal)
- Punctuation

\[\text{Diagram: Dependency Grammar 17(29)}\]
A dependency structure can be defined as a directed graph $G$, consisting of

- a set $V$ of nodes,
- a set $E$ of arcs (edges),
- a linear precedence order $<$ on $V$ (not in every theory)

Labeled graphs:

- Nodes in $V$ are labeled with word forms (and annotation).
- Arcs in $E$ are labeled with dependency types.

Notational conventions ($i, j \in V$):

- $i \rightarrow j \equiv (i, j) \in E$
- $i \rightarrow^* j \equiv i = j \lor \exists k : i \rightarrow k, k \rightarrow^* j$
Formal Conditions on Dependency Graphs

- Intuitions:
  - Syntactic structure is complete (Connectedness).
  - Syntactic structure is hierarchical (Acyclicity).
  - Every word has at most one syntactic head (Single-Head).
- Connectedness can be enforced by adding a special root node.

**Economic news had little effect on financial markets.**
Formal Conditions on Dependency Graphs

Intuitions:

- Syntactic structure is complete (Connectedness).
- Syntactic structure is hierarchical (Acyclicity).
- Every word has at most one syntactic head (Single-Head).

Connectedness can be enforced by adding a special root node.

```
root Economic news had little effect on financial markets.
```
Formal Conditions on Dependency Graphs

- $G$ is (weakly) connected:
  - For every node $i$ there is a node $j$ such that $i \rightarrow j$ or $j \rightarrow i$.

- $G$ is acyclic:
  - If $i \rightarrow j$ then not $j \rightarrow^* i$.

- $G$ obeys the single-head constraint:
  - If $i \rightarrow j$, then not $k \rightarrow j$, for any $k \neq i$.

- $G$ is projective:
  - If $i \rightarrow j$ then $i \rightarrow^* k$, for any $k$ such that $i < k < j$ or $j < k < i$.
Projectivity

Projectivity (or, less commonly, adjacency [Hudson(1990)])

- A head (A) and a dependent (B) must be adjacent: A is adjacent to B provided that every word between A and B is a subordinate of A.

(2) with great difficulty
(3) *great with difficulty

- with $\rightarrow$ difficulty
- difficulty $\rightarrow$ great

*great with difficulty* is ruled out because branches would have to cross in that case
Projectivity

► Most theoretical frameworks do not assume projectivity.
► Non-projective structures are needed to account for
  ► long-distance dependencies,
  ► free word order.
Valency and Grammaticality

An important concept in many variants of DG is that of *valency* = the ability of a word to take arguments

A lexicon might look like the following

[Hajič et al.(2003)Hajič, Panevová, Urešová, Bémová, Kolářová and Pajas]:

<table>
<thead>
<tr>
<th></th>
<th>Slot$_1$</th>
<th>Slot$_2$</th>
<th>Slot$_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$sink_1$</td>
<td>ACT(nom)</td>
<td>PAT(acc)</td>
<td></td>
</tr>
<tr>
<td>$sink_2$</td>
<td>PAT(nom)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>give</td>
<td>ACT(nom)</td>
<td>PAT(acc)</td>
<td>ADDR(dat)</td>
</tr>
</tbody>
</table>

To determine grammaticality (roughly) ...

1. Words have valency requirements that must be satisfied
2. Apply general rules to the valencies to see if a sentence is valid
Capturing Adjuncts and Complements

There are two main kinds of dependencies for $A \rightarrow B$:

- **Head-Complement**: if $A$ (the head) has a slot for $B$, then $B$ is a complement
- **Head-Adjunct**: if $B$ has a slot for $A$ (the head), then $B$ is an adjunct

$B$ is dependent on $A$ in either case, but the selector is different

- The adjunct/complement distinction is captured in the type of dependency relation and/or in the lexicon
Layers of dependencies

[Mel’čuk(1988)] allows for different dependency layers

It looks like a subject depends on the verb, but the form of the verb depends on the subject (mutual dependence):

(4) a. The child is playing.
    b. The children are playing.

Solution:

- Dependence of *child/children* on the verb is syntactic
- Dependence of the verb(form) on the subject is morphological
Double dependencies

Likewise, here it seems that *clean* depends both on the verb *wash* and on the noun *dish*.

(5) Wash the dish *clean*.

Solution:

- Dependence of *clean* on *wash* is syntactic (cf. case)
- Dependence of *clean* on *dish* is semantic (cf. gender)

(6) My našli zal pust-ym
    We found the hall$_{masc}$ empty$_{masc.sg.inst}$
Double dependencies (2)

Hudson’s Word Grammar [Hudson(2004)] explicitly allows for structure-sharing, explicitly violating the single-head constraint:

- wash $\rightarrow$ clean
- dish $\rightarrow$ clean

NB: Hudson also uses this to account for non-projectivity
Relation to phrase structure

What is the relation between DG and PSG?

- If a PS tree has heads marked, then you can derive the dependencies
- Likewise, a DG tree can be converted into a PS tree by grouping a word with its dependents
  - But what the constituents are is still open (binary-branching, flat)
  - And phrases are not categorized
Advantages and Disadvantages of DG

Advantages:

► Close connection to semantic representation
► More flexible structure for, e.g., non-constituent coordination
► Easier to capture some typological regularities
► Vast & expanding body of computational work on dependency parsing

Disadvantages:

► No constituents makes analyzing coordination difficult
► No distinction between modifying a constituent vs. an individual word
► Harder to capture things like, e.g., subject-object asymmetries


References

In Vilmos Agel, Ludwig M. Eichinger, Hans-Werner Eroms, Peter Hellwig, Hans Jürgen Heringer and Hening Lobin (eds.), *Dependency and Valency*, Walter de Gruyter, pp. 593–635.


