Syntactic Analysis
Daniel Zeman

December 14, 2022
1. Constituents vs. Dependencies

2. Universal Dependencies
   - A Tour through UD Syntax
   - Nonverbal Predicate and Copula
   - Core Arguments vs. Oblique Dependents
   - Ellipsis and Enhanced UD
Constituents vs. Dependencies
Syntactic Structure

- Different shapes in different theories
- Typically a tree
  - Constituents (phrase tree structure)
  - Dependencies (dependency tree structure)
Constituent Tree

\[
(S \ (NP \ (N \ Paul)) \ (VP \ (V \ gave) \ (NP \ (N \ Peter)) \ (NP \ (C \ two) \ (N \ pears))))
\]
Paul gave Peter two pears.
[#,0] ([gave,2] ([Paul,1], [Peter,3], [pears,5] ([two,4])), [.,6])

- # / AuxS
- gave / Pred
- Paul / Sb
- Peter / Obj
- pears / Obj
- two / Atr
Paul gave Peter two pears.
Constituents vs. Dependencies

• The two models are interconnected

• Sentence divided to phrases (constituents)
  • Recursive: phrases divided to smaller phrases
  • The smallest phrases are words

• There are dependencies (relations) between words (constituents)
  • Head of phrase = governing node, parent node
  • The other nodes are dependent nodes, children of the head
Phrase vs. Dependency Trees

S
  └── NP
       └── N
            └── Paul
  └── VP
      └── NP
            └── N
                 └── Peter
      └── NP
            └── N
                 └── two
                 └── N
                      └── pears

N V NP NP
  └── gave
      └── Paul
  └── C
      └── Peter
  └── N
      └── two
      └── N
           └── pears

Paul gave Peter two pears

root
  └── obj
      └── nummod
        └── two
        └── pears
  └── nsubj
      └── nsubj
          └── Paul
dep
  └── obj
      └── Peter
  └── nummod
      └── two
      └── pears

Universal Dependencies
Phrase vs. Dependency Trees

- Phrase trees
  - Usually do not mark the head
  - May not mark the function of the constituent in the superordinate constituent
Phrase vs. Dependency Trees

• Phrase trees
  • Usually do not mark the head
  • May not mark the function of the constituent in the superordinate constituent

• Dependency trees
  • Do not show nonterminals (phrase types)
    • Nor any other phrase-level features
  • Do not show “how the sentence is generated” (order, recursion, proximity of constituents)
Example

Constituents vs. Dependencies

Universal Dependencies
Discontinuous Phrases

- Classical context-free grammar cannot describe them!
- They cannot be represented by bracketing.
- **English example:** *I found the best example ever.*
- **Czech example:** *(Soubor (se nepodařilo) otevřít).* “File couldn’t be opened.”

```
VP(nepodařilo)

VR(nepodařilo)  VP_{inf}(otevřít)

T  V  V_{inf}  N

se  nepodařilo  otevřít  soubor
```
Nonprojectivity

• Dependency tree including word order (x-coordinate of nodes).
• Projection to the base: the vertical from the node crosses a dependency (nonprojective edge).
• Formally:
  • Dependency \([g, x_g], [d, x_d]\) where \(x_w\) is the order of the word \(w\) in the sentence.
  • There exists a node \([n, x_n]\) that \(x_g < x_n < x_d\) or \(x_d < x_n < x_g\) and \([n, x_n]\) is not in subtree rooted by \([g, x_g]\).
• Informally: The string spanned by the subtree of the governing node is discontinuous, it contains gaps.
Nonprojectivity Can Be Handled by a Dependency Tree!

```
  nepodařilo / Pred
     /   
  se / AuxT  otevřít / Sb
     /   
  soubor / Obj
```
Nonprojectivity Can Be Handled by a Dependency Tree!

soubor file
se itself	nepodařilo did-not-succeed
otevřít to-open

root

Obj

AuxT

Sb
Universal Dependencies
Outline

1 Constituents vs. Dependencies

2 Universal Dependencies
   - A Tour through UD Syntax
   - Nonverbal Predicate and Copula
   - Core Arguments vs. Oblique Dependents
   - Ellipsis and Enhanced UD
Constituents vs. Dependencies
My daughter bought some bread and cheese

Min datter købte nogle brød og ost

Min dotter köpte några bröd och ost
Universal Dependencies

http://universaldependencies.org/

Milestones:

- 2008-05 Interset (morphological features)
- 2012-05 Google Universal POS tags
- 2012-05 HamleDT (harmonized Prague-style treebanks)
- 2013-08 Google Universal Dependency Treebank
- 2014-05 Universal Stanford Dependencies
- 2014-04 EACL Göteborg, kick-off meeting of UD
- 2014-10 UD guidelines version 1
- 2015-01 released first 10 treebanks
- every ~6 months new release
- 2016-12 UD guidelines version 2
- 2017-05 CoNLL Shared Task in parsing UD
- 2018-06 second Shared Task
- every ~6 months new release
• Same things annotated same way across languages...
• ... while highlighting different coding strategies
Manning’s Law

The secret to understanding UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD must be satisfactory on linguistic analysis grounds for individual languages.
2. UD must be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
3. UD must be suitable for rapid, consistent annotation by a human annotator.
4. UD must be easily comprehended and used by a non-linguist, whether a language learner or an engineer with prosaic needs for language processing. … it leads us to favor traditional grammar notions and terminology.
5. UD must be suitable for computer parsing with high accuracy.
6. UD must support well downstream language understanding tasks (relation extraction, reading comprehension, machine translation, …)

It’s easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.
Manning’s Law

The secret to understanding UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD must be satisfactory on linguistic analysis grounds for individual languages.

2. UD must be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.

It’s easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.
Manning’s Law

The secret to understanding UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD must be satisfactory on linguistic analysis grounds for individual languages.
2. UD must be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
3. UD must be suitable for rapid, consistent annotation by a human annotator.

It’s easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.
Manning’s Law

The secret to understanding UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD must be satisfactory on linguistic analysis grounds for individual languages.
2. UD must be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
3. UD must be suitable for rapid, consistent annotation by a human annotator.
4. UD must be easily comprehended and used by a non-linguist, whether a language learner or an engineer with prosaic needs for language processing. ... it leads us to favor traditional grammar notions and terminology.

It’s easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.
Manning’s Law

The secret to understanding UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD must be satisfactory on linguistic analysis grounds for individual languages.
2. UD must be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
3. UD must be suitable for rapid, consistent annotation by a human annotator.
4. UD must be easily comprehended and used by a non-linguist, whether a language learner or an engineer with prosaic needs for language processing. ... it leads us to favor traditional grammar notions and terminology.
5. UD must be suitable for computer parsing with high accuracy.

It’s easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.
Manning’s Law

The secret to understanding UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD must be satisfactory on linguistic analysis grounds for individual languages.
2. UD must be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
3. UD must be suitable for rapid, consistent annotation by a human annotator.
4. UD must be easily comprehended and used by a non-linguist, whether a language learner or an engineer with prosaic needs for language processing. ... it leads us to favor traditional grammar notions and terminology.
5. UD must be suitable for computer parsing with high accuracy.
6. UD must support well downstream language understanding tasks (relation extraction, reading comprehension, machine translation, ...)

It’s easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.
Same Thing Same Way

George killed the dragon

PROPN VERB DET NOUN
Same Thing Same Way

- George killed the dragon
- Mharaigh Seoirse an dragan

Constituents vs. Dependencies
Same Thing Same Way

1. **George killed the dragon**
   - **George**: PROPN
   - **killed**: VERB
   - **the**: DET
   - **dragon**: NOUN

2. **Mharaigh Seoirse an dragan**
   - **Mharaigh**: PROPN
   - **Seoirse**: PROPN
   - **an**: DET
   - **dragon**: NOUN

3. **Jorge mató al dragón**
   - **Jorge**: PROPN
   - **mató**: VERB
   - **al**: ???
   - **dragón**: NOUN

**Constituents vs. Dependencies**

23/110
Same Thing Same Way

Constituents vs. Dependencies
Same Thing Same Way
Same Meaning ≠ Same Construction!

Constituents vs. Dependencies
Same Meaning ≠ Same Construction!

He killed the dragon

The dragon was killed by him

Constituents vs. Dependencies
Same Meaning ≠ Same Construction!

He killed the dragon

The dragon was killed by him

His killing of the dragon
Same Meaning ≠ Same Construction!

Constituents vs. Dependencies
राजा विष्णुशर्मणां आहूया प्रोवाचा
king Vishnusharma having-summoned said

VerbForm=Conv VerbForm=Fin
The king summoned Vishnusharma and said.

- **Subject (nsubj)**: राजा rājā (king)
- **Object (obj)**: विष्णुशर्माः viṣṇuśarmāḥ (Vishnusharma)
- **Adverb (advcl)**: आहूयā āhūya (having-summoned)
- **Verb (root)**: प्रोवाच provāca (said)
Universal Dependencies
A Tour through UD Syntax
Outline

1. Constituents vs. Dependencies

2. Universal Dependencies
   - A Tour through UD Syntax
   - Nonverbal Predicate and Copula
   - Core Arguments vs. Oblique Dependents
   - Ellipsis and Enhanced UD
The cat could have chased all the dogs down the street.

```
DET NOUN AUX AUX VERB DET DET NOUN ADP DET NOUN PUNCT
```
Syntax

The cat could have chased all the dogs down the street.

- Content words are related by dependency relations

Constituents vs. Dependencies
The cat could have chased all the dogs down the street.

- Content words are related by dependency relations
- Function words attach to closest content words
The cat could have chased all the dogs down the street.

- Content words are related by dependency relations
- Function words attach to closest content words
- Punctuation attach to head of phrase or clause
The cat could have chased all the dogs down the street.

Not "dependency" in the strictly syntactic sense!
The dog was chased by the cat.
The dog was chased by the cat.
The dog was chased by the cat.

Кучето беше преследвано от котката.
The dog was chased by the cat.
## Dependents of Clauses (Verbal or Not)

<table>
<thead>
<tr>
<th>Core</th>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>nsubj</td>
<td>csubj</td>
<td>advmod</td>
<td>aux</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>advcl</td>
<td>discourse</td>
</tr>
<tr>
<td>Non-Core</td>
<td>obl</td>
<td>advcl</td>
<td></td>
<td>aux</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>expl</td>
<td>cop</td>
</tr>
</tbody>
</table>

## Dependents of Verbs, Adjectives and Adverbs

<table>
<thead>
<tr>
<th>Core</th>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>obj</td>
<td>ccomp</td>
<td></td>
</tr>
<tr>
<td></td>
<td>iobj</td>
<td>xcomp</td>
<td></td>
</tr>
<tr>
<td>Non-Core</td>
<td>obl</td>
<td>advcl</td>
<td>advmod</td>
</tr>
<tr>
<td></td>
<td>expl</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Dependents of Nominals

<table>
<thead>
<tr>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>nmod</td>
<td>acl</td>
<td>amod</td>
<td>det</td>
</tr>
<tr>
<td>appos</td>
<td></td>
<td>nummod</td>
<td>case</td>
</tr>
</tbody>
</table>
Dependents of Nominals

<table>
<thead>
<tr>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>nmod</td>
<td>acl</td>
<td>amod</td>
<td>det</td>
</tr>
<tr>
<td>appos</td>
<td></td>
<td>nummod</td>
<td>case</td>
</tr>
<tr>
<td>compound</td>
<td></td>
<td></td>
<td>clf</td>
</tr>
<tr>
<td>flat</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Constituents vs. Dependencies
• Coordinate structures are headed by the first conjunct
  • Subsequent conjuncts depend on it via the conj relation
  • Conjunctions depend on the next conjunct via the cc relation
  • Punctuation marks depend on the next conjunct via the punct relation
But Some Languages Might Prefer the Opposite

- Coordinate structures would be headed by the last conjunct
  - Preceding conjuncts would depend on it via the `conj` relation
  - Conjunctions would depend on the preceding conjunct
  - Punctuation marks would depend on the preceding conjunct
## Multiword Expressions

<table>
<thead>
<tr>
<th>Relation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixed</td>
<td><em>in spite of</em>, <em>as well as</em>, <em>ad hoc</em></td>
</tr>
<tr>
<td>flat</td>
<td><em>president Havel</em>, <em>New York</em>, <em>four thousand</em></td>
</tr>
<tr>
<td>compound</td>
<td><em>phone book</em>, <em>dress up</em></td>
</tr>
<tr>
<td>goeswith</td>
<td><em>notwith standing</em>, <em>with out</em></td>
</tr>
</tbody>
</table>

- UD annotation **almost** does not permit “words with spaces”
  - Multiword expressions are analyzed using special relations
  - The **fixed**, **flat** and **goeswith** relations are always head-initial
  - The **compound** relation reflects the internal structure
- Words with spaces may be allowed in v2:
  - Vietnamese (spaces delimit syllables, not words)
  - Numbers (“1 000 000”)
  - Possibly other approved cases, e.g. multi-word abbreviations
<table>
<thead>
<tr>
<th>Relation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>parataxis</td>
<td>Loosely linked clauses of same rank</td>
</tr>
<tr>
<td>list</td>
<td>Lists without syntactic structure</td>
</tr>
<tr>
<td>orphan</td>
<td>Orphans in ellipsis linked together</td>
</tr>
<tr>
<td>reparandum</td>
<td>Disfluency linked to (speech) repair</td>
</tr>
<tr>
<td>foreign</td>
<td>Elements within opaque stretches of code switching</td>
</tr>
<tr>
<td>dep</td>
<td>Unspecified dependency</td>
</tr>
<tr>
<td>root</td>
<td>Syntactically independent element of clause/phrase</td>
</tr>
</tbody>
</table>
Language-specific Relation Subtypes

- Language-specific relations are **subtypes** of universal relations added to capture important phenomena
- Subtyping permits us to “back off” to universal relations

### Language-specific Relation Subtypes

<table>
<thead>
<tr>
<th>Relation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>acl:relcl</td>
<td>Relative clause</td>
</tr>
<tr>
<td>compound:prt</td>
<td>Verb particle (dress <strong>up</strong>)</td>
</tr>
<tr>
<td>nmod:poss</td>
<td>Possessive nominal (<strong>Mary</strong>’s book)</td>
</tr>
<tr>
<td>obl:agent</td>
<td>Agent in passive (saved <strong>by the bell</strong>)</td>
</tr>
<tr>
<td>cc:preconj</td>
<td>Preconjunction (<strong>both</strong> ... <strong>and</strong>)</td>
</tr>
<tr>
<td>det:predet</td>
<td>Predeterminer (<strong>all</strong> those ...)</td>
</tr>
</tbody>
</table>
Universal Dependencies

Nonverbal Predicate and Copula
Outline

1. Constituents vs. Dependencies

2. Universal Dependencies
   - A Tour through UD Syntax
   - Nonverbal Predicate and Copula
   - Core Arguments vs. Oblique Dependents
   - Ellipsis and Enhanced UD
Nonverbal Predicate and Copula

- Some languages use a copula verb:
  
  - \text{Ivan} \text{ is } \text{the best dancer}.

- Some languages use a copula pronoun:
  
  - \text{Ivan} – to \text{ najlepszy tancerz}.
  - \text{Ivan} – it \text{ best dancer}.
Nonverbal Predicate and Copula

• Some languages use a copula verb:

  Ivan is the best dancer.

• Some languages omit the copula:

  Иван лучший танцор.

  Ivan lučšij tancor.
  Ivan best dancer.
Some languages use a copula verb:

Ivan was the best dancer.

Some languages use it only in some tenses:

Ivan был лучшим танцором.
Copula Verbs: We Are Restrictive!

- *To be* is copula:

  Ivan is the best dancer.

- *To become* is not copula:

  Ivan became the best dancer.
Once Copula, Always Copula!

• This is parallel with Russian:

Ivan is the best dancer.

• This is also parallel with Russian:

Ivan is today in Moscow.
Well, Almost...

- This is parallel with Russian:
  
  Ivan is today in Moscow.

- But not with this in English:
  
  There is a dancer in Moscow.
Clauses and Copula

- A clause can be the subject:
  
  ```
  The problem is that he is missing .
  ```

- But it cannot be annotated as the nonverbal predicate:
  
  ```
  The problem is that he is missing .
  ```
Universal Dependencies
Core Arguments vs. Oblique Dependents
Outline

1. Constituents vs. Dependencies

2. Universal Dependencies
   - A Tour through UD Syntax
   - Nonverbal Predicate and Copula
   - Core Arguments vs. Oblique Dependents
   - Ellipsis and Enhanced UD
### Dependents of Clauses (Verbal or Not)

<table>
<thead>
<tr>
<th>Core</th>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>nsubj</td>
<td>csubj</td>
<td>advmod</td>
<td>aux</td>
</tr>
<tr>
<td></td>
<td>obl</td>
<td>advcl</td>
<td>discourse</td>
<td>cop</td>
</tr>
<tr>
<td></td>
<td>vocative</td>
<td>dislocated</td>
<td></td>
<td>mark</td>
</tr>
<tr>
<td>Non-Core</td>
<td>expl</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Dependents of Verbs, Adjectives and Adverbs

<table>
<thead>
<tr>
<th>Core</th>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>obj</td>
<td>ccomp</td>
<td></td>
</tr>
<tr>
<td></td>
<td>iobj</td>
<td>xcomp</td>
<td></td>
</tr>
<tr>
<td>Non-Core</td>
<td>obl</td>
<td>advcl</td>
<td>advmod</td>
</tr>
<tr>
<td></td>
<td>expl</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Dependents of Nominals

<table>
<thead>
<tr>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>nmod</td>
<td>acl</td>
<td>amod</td>
<td>det</td>
</tr>
<tr>
<td>appos</td>
<td></td>
<td>nummod</td>
<td>case</td>
</tr>
</tbody>
</table>
I gave her a book.
Information Packaging

Constituents vs. Dependencies
He loaded the wagon with hay.

Constituents vs. Dependencies

Universal Dependencies
He loaded the wagon with hay.

He loaded hay on the wagon.
UD is NOT about Semantic Roles!

Constituents vs. Dependencies
Manning’s Law – What If We Do Semantic Roles?

The secret to understanding the design and current success of UD is to realize that the design is a very subtle compromise between approximately 6 things:

1. UD must be satisfactory on linguistic analysis grounds for individual languages.
2. UD must be good for linguistic typology, i.e., providing a suitable basis for bringing out cross-linguistic parallelism across languages and language families.
3. UD must be suitable for rapid, consistent annotation by a human annotator.
4. UD must be easily comprehended and used by a non-linguist, whether a language learner or an engineer with prosaic needs for language processing. ... it leads us to favor traditional grammar notions and terminology.
5. UD must be suitable for computer parsing with high accuracy.
6. UD must support well downstream language understanding tasks (relation extraction, reading comprehension, machine translation, ...)

It’s easy to come up with a proposal that improves UD on one of these dimensions. The interesting and difficult part is to improve UD while remaining sensitive to all these dimensions.
UD Avoids Argument-Adjunct Distinction!

Constituents vs. Dependencies
Avoiding an Argument-Adjunct Distinction

• From the guidelines:
  • Subtle, unclear, and frequently argued over
  • Questionable as a categorical distinction
  • Best practical solution is to eliminate it

BUT:
• Cannot be eliminated completely
• Some people/data have it and want to keep it
• It aligns well with traditional grammars
⇒ there is now a relation subtype obl:arg

AND
• I will argue that Core-oblique distinction is unclear and argued over too
• (Though I will not propose to discard it.)

Constituents vs. Dependencies
Avoiding an Argument-Adjunct Distinction

• From the guidelines:
  • Subtle, unclear, and frequently argued over
  • Questionable as a categorical distinction
  • Best practical solution is to eliminate it

• BUT:
  • Cannot be eliminated completely
  • Some people/data have it and want to keep it
    • It aligns well with traditional grammars
  • ⇒ there is now a relation subtype obl:arg
Avoiding an Argument-Adjunct Distinction

• From the guidelines:
  • Subtle, unclear, and frequently argued over
  • Questionable as a categorical distinction
  • Best practical solution is to eliminate it

• BUT:
  • Cannot be eliminated completely
  • Some people/data have it and want to keep it
    • It aligns well with traditional grammars
  • ⇒ there is now a relation subtype obl:arg

• AND I will argue that
  • Core-oblique distinction is unclear and argued over too
  • (Though I will not propose to discard it.)
So What Is Core and Why?

Constituents vs. Dependencies  Universal Dependencies
Community Confusion

- UD v1 guidelines took core-oblique for granted

- English (simplified):
  - Bare noun phrase \( \Rightarrow \) core argument (nsubj, obj, iobj)
  - Prepositional phrase \( \Rightarrow \) oblique argument or adjunct (obl)

- Other languages: not necessarily! (Spanish, Japanese)

- But some people simply took the English rule…

- Clashing with traditional terminology

- Grammars of German, Czech etc. define prepositional objects

- But these are not necessarily core…

- Yet some people took their national definition of object…
Community Confusion

• UD v1 guidelines took core-oblique for granted

• English (simplified):
  • Bare noun phrase ⇒ core argument (nsubj, obj, iobj)
  • Prepositional phrase ⇒ oblique argument or adjunct (obl)

• Other languages: not necessarily! (Spanish, Japanese)
  • But some people simply took the English rule...
  • Manning’s law: non-linguists!
Community Confusion

• UD v1 guidelines took core-oblique for granted

• English (simplified):
  • Bare noun phrase ⇒ core argument (nsubj, obj, iobj)
  • Prepositional phrase ⇒ oblique argument or adjunct (obl)

• Other languages: not necessarily! (Spanish, Japanese)
  • But some people simply took the English rule...
  • Manning’s law: non-linguists!

• Clash with traditional terminology
  • Grammars of German, Czech etc. define prepositional objects
  • But these are not necessarily core...
  • Yet some people took their national definition of object...
Language-specific Coding Strategy

• Idea:
  • Oblique arguments are marked similarly to adjuncts (prepositions, certain morphological cases...)
  • Core arguments are marked differently
    • ⇒ easy for annotators and non-linguists!

• Why are core arguments special?
  • They tend to be targeted by grammatical rules
    • Passivization
    • Control verbs
    • Reflexives
    • ...

Constituents vs. Dependencies
• Core vs. oblique is not defined in traditional grammar
• How shall we define it?

Andrews, 2007 (In Shopen: Language Typology)

Identify primary transitive predicates
• We need semantic roles for this! (One-time only.)

Actor/agent = function $A$

Undergoer/patient = function $P$

• Note the way they are coded
• Note other grammatical rules that target them
• Generalize to other predicates with same coding and rules

Then define:
• function $A$ $\Rightarrow$ nsubj
• function $P$ $\Rightarrow$ obj
Language-specific Coding Strategy

• Core vs. oblique is not defined in traditional grammar
• How shall we define it?

• Andrews, 2007 (In Shopen: Language Typology)
  • Identify primary transitive predicates
    • We need semantic roles for this! (One-time only.)
Language-specific Coding Strategy

- Core vs. oblique is not defined in traditional grammar
- How shall we define it?

Andrews, 2007 (In Shopen: Language Typology)
  - Identify primary transitive predicates
    - We need semantic roles for this! (One-time only.)
  - Actor/agent = function $A$
  - Undergoer/patient = function $P$
Language-specific Coding Strategy

- Core vs. oblique is not defined in traditional grammar
- How shall we define it?

- Andrews, 2007 (In Shopen: Language Typology)
  - Identify primary transitive predicates
    - We need semantic roles for this! (One-time only.)
  - Actor/agent = function A
  - Undergoer/patient = function P
  - Note the way they are coded
  - Note other grammatical rules that target them
  - Generalize to other predicates with same coding and rules
Language-specific Coding Strategy

• Core vs. oblique is not defined in traditional grammar
• How shall we define it?

• Andrews, 2007 (In Shopen: Language Typology)
  • Identify primary transitive predicates
    • We need semantic roles for this! (One-time only.)
  • Actor/agent = function A
  • Undergoer/patient = function P
  • Note the way they are coded
  • Note other grammatical rules that target them
  • Generalize to other predicates with same coding and rules

• Then define:
  • function A ⇒ nsubj
  • function P ⇒ obj
Transitive Predicates in English

John \textit{kills} Mary \textit{(primary transitive)}

PROPN VERB PROPN

\textit{(primary transitive)}
**Transitive Predicates in English**

- **John** kills **Mary** *(primary transitive)*

- **John** loves **Mary** *(generalized transitive)*
Transitive Predicates in English

John kills Mary (primary transitive)

John loves Mary (generalized transitive)
Transitive Predicates in English

- nominal
  - Case=Nom
  - bare NP
  - pre-verb
  - cross-ref on verb

- VERB
  - Voice=Act(Pass)
  - declarative clause
  - agreement

- nominal
  - Case=Acc
  - bare NP
  - post-verb

Constituents vs. Dependencies
Passivization in English

Constituents vs. Dependencies
Subject Control in English

Constituents vs. Dependencies
Object Control in English

Constituents vs. Dependencies
Some Problems

• Some temporal adjuncts are bare noun phrases
  • I work the whole week.
  • I work every Friday.

• At least it cannot passivize:
  • *The whole week is worked by me.
  • *Every Friday is worked by me.

• But…
Some Problems

• Some transitive verbs cannot passivize
  • *John has a new car.
  • *A new car is had by John.
  • Friday does not suit me.
  • *I am not suited by Friday.
Some Problems

• Some transitive verbs cannot passivize
  • *John has a new car.
  • *A new car is had by John.
  • Friday does not suit me.
  • *I am not suited by Friday.

• Some prepositional verbs can passivize
  • You can rely on Ben.
    • Ben can be relied on.
  • They will take care of your children.
    • Your children will be taken care of.
Tentative Summary?

- The borderline is inherently fuzzy
- No universally applicable and exact algorithm
- Better described in terms of probability
Tentative Summary?

- The borderline is inherently fuzzy
- No universally applicable and exact algorithm
- Better described in terms of probability

- Core coding *not favored by adjuncts*
- Oblique coding *similar to most adjuncts*
- Passivization etc. may help...
- ... but does *not* work as *strict criterion*
Tentative Summary?

- The borderline is inherently fuzzy
- No universally applicable and exact algorithm
- Better described in terms of probability
- Core coding not favored by adjuncts
- Oblique coding similar to most adjuncts
- Passivization etc. may help…
- … but does not work as strict criterion
- Semantic roles needed when starting a new language
- Argument-adjunct needed to describe exceptions (*the whole week*)
Intransitive Predicates

• Just one core argument
  • We already “know” how to find out if there are two

• \( \Rightarrow \) function \( S \)
  • Regardless of semantic role:
    • *John runs.*
    • *John sleeps.*
    • *John falls.*

• Then define:
  • function \( S \Rightarrow nsubj \)
Ditransitive Predicates

- Three core arguments
- Is one of them “least core”? \( \Rightarrow \) iobj
- (Alternatively, we could look at the semantic roles once again.)

```
I gave her a book
PRON VERB PRON DET NOUN
```

- Passivization:
  - *She* was given *a book* by *me*.
  - *?A book* was given *her* by *me*.
Ditransitive Predicates

- Three core arguments
- Is one of them “least core”? ⇒ iobj
- (Alternatively, we could look at the semantic roles once again.)

Andrews (2007): the status of the notion of ‘indirect object’ is problematic and difficult to sort out. The top priority is to work out what properties recipients and themes do and do not share with P arguments of primary transitive verbs.
Jorge mató al dragón

El dragón fue matado por Jorge
Spanish Transitive Clauses

- nominal
  - Case=Nom
  - bare NP
  - pre-verb
  - cross-ref on verb

- VERB
  - Voice=Act(,Pass)
  - declarative clause
  - agreement

- (ADP)

- nominal
  - Case=Acc
  - (or bare NP)
  - post-verb

Constituents vs. Dependencies
Spanish Adjunct Exceptions

El trabaja toda la semana
He works whole the week

Subiremos a el tren a las cinco
We-will-board to the train at the five
Spanish Ditransitive Clauses

Constituents vs. Dependencies
Spanish Ditransitive Clauses

Pedro le dio un libro a Isabel

PROPN PRON VERB DET NOUN ADP PROPN

Un libro fue dado a Isabel por Pedro

DET NOUN AUX VERB ADP PROPN ADP PROPN

Constituents vs. Dependencies
Jiří zabil draka.

Drak byl zabit Jiřím.
Czech Transitive Clauses

Constituents vs. Dependencies

---

- **nominal**: Case=Nom, bare NP, pre-verb, cross-ref on verb
- **VERB**: Voice=Act(,Pass), declarative clause, agreement
- **nominal**: Case=Acc, bare NP, post-verb
Czech Adjunct Exceptions

Pracuje celý týden

He works whole week

Constituents vs. Dependencies
Czech Ditransitive Clauses

Constituents vs. Dependencies
Dative: Recipient vs. Beneficiary

- Petr caption below the sentence.
- četl caption below the sentence.
- knihu caption below the sentence.
- read caption below the sentence.
- to-Katka caption below the sentence.
- book caption below the sentence.
- zlomil caption below the sentence.
- nohu caption below the sentence.
- broke caption below the sentence.
- Katka's caption below the sentence.
- leg caption below the sentence.

Constituents vs. Dependencies

Universal Dependencies

76/110
Monotransitive with Dative?

- Zuzka pomohla Martinovi s úkolem
- Martinovi bylo pomůženo s úkolem

**Constituents vs. Dependencies**

- Dependencies: root [nsubj:Zuzka] [obj:pomohla] [case:s úkolem]
Monotransitive with Genitive?

"Novináři musí dbát zásad objektivity."

"Journalists must observe principles of objectivity."

Constituents vs. Dependencies
Monotransitive with Instrumental?

Karel hýbal nábytkem

PROPN VERB NOUN
Case=Nom Case=Ins

Nábytkem bylo hýbáno

NOUN AUX ADJ
Case=Ins Gender=Neut Number=Sing
They relied on director's decision.
• There is a core-oblique scale:
  - Nom > Acc > Gen, Dat > Ins > preposition

• Where is the borderline?
• There is a core-oblique scale:
  • Nom > Acc > Gen,Dat > Ins > preposition

• Where is the borderline?

• UD Czech 1.0: object = argument
  • Nom, Acc, Gen, Dat, Ins, ADP > “adverbial”
There is a core-oblique scale:

- **Nom** > **Acc** > **Gen, Dat** > **Ins** > **preposition**

Where is the borderline?

- UD Czech 1.0: object = argument
  - Nom, Acc, Gen, Dat, Ins, ADP > “adverbial”
- UD Czech 2.1–2.5: bare NP > PP
  - Nom, Acc, Gen, Dat, Ins > ADP + adjuncts
• There is a core-oblique scale:
  • Nom > Acc > Gen, Dat > Ins > preposition

• Where is the borderline?

• UD Czech 1.0: object = argument
  • Nom, Acc, Gen, Dat, Ins, ADP > “adverbial”

• UD Czech 2.1–2.5: bare NP > PP
  • Nom, Acc, Gen, Dat, Ins > ADP + adjuncts

• UD Czech 2.6 (May 2020):
  • Nom, Acc > Gen, Dat, Ins, ADP + adjuncts
• There is a core-oblique scale:
  • **Nom > Acc > Gen, Dat > Ins > preposition**

• Where is the borderline?

• UD Czech 1.0: object = argument
  • Nom, Acc, Gen, Dat, Ins, ADP > “adverbial”

• UD Czech 2.1–2.5: bare NP > PP
  • Nom, Acc, Gen, Dat, Ins > ADP + adjuncts

• UD Czech 2.6 (May 2020):
  • Nom, Acc > Gen, Dat, Ins, ADP + adjuncts
  • ⇒ No ditransitives in Czech!
  • (Exception: *učit* “to teach” takes two Acc.)
Basque Transitive Clauses

- Nominal Case=Erg
- Nominal Case=Erg
- Nominal Case=Dat
- Bare NP
- Pre-verb
- Cross-ref on verb

VERB
- Declarative clause
- Agreement

Nominal Case=Abs
- Nominal Case=Dat
- Nominal Case=Abs
- Bare NP
- Post-verb
- Cross-ref on verb

Constituents vs. Dependencies
Basque Transitive Clauses

Constituents vs. Dependencies

83/110
Basque Intransitive Clauses

Gizona hil da
The-man died it-has
NOUN VERB AUX
Case=Abs

Urak irakin du
Water boiled it-has-it
NOUN VERB AUX
Case=Erg

Constituents vs. Dependencies
Basque Ditransitive Clauses

Constituents vs. Dependencies

Universal Dependencies
Basque Ditransitive Clauses

Constituents vs. Dependencies
Basque Causative Applied to Dative Subject

Zopa
Soup
NOUN
Case=Abs

izugarri
greatly
ADV

gustatzen
pleasing
VERB

zaio
it-is-it
AUX

mutilari
to-boy
NOUN
Case=Dat

Goseak
Hunger
NOUN
Case=Erg

zopa
soup
NOUN
Case=Abs

izugarri
greatly
ADV

gustatuerazi
made-pleasing
VERB

zion
it-has-it-it
AUX

mutilari
to-boy
NOUN
Case=Dat
Yidiŋ Transitive Clauses

Constituents vs. Dependencies
“I, (who) was slapped by the woman, laughed”

Constituents vs. Dependencies

Universal Dependencies

Case=Nom
Case=Acc
Case=Erg
"I, (who) was slapped by the woman, laughed"

The coreferential (and elidable) NP must have S or P function.
“I, (who) was laughing, was slapped by the woman”

The coreferential (and elidable) NP must have S or P function.
Yidiɲ Antipassive

“I, (who) was slapping the woman, laughed”

Original P is now oblique and original A is now S.
Tagalog Transitive Clauses

```
VERB DET/ADP nominal
  ang-NP Case=Nom?

DET/ADP nominal
  ng-NP Case=Acc?
```
Magaalis ang babae ng sa sako
Will-take the woman rice from sack

Aalisin ng babae ang bigas sa sako
Will-take woman the rice from sack

Constituents vs. Dependencies
Tagalog Locative Voice ⇒ Ditransitive!

Aalisan ng babae ng bigas ang sako

Will-take woman rice from-the sack

Voice=Locf Case=Acc Case=Acc Case=Nom

VERB DET NOUN DET NOUN DET NOUN
Tagalog Benefactive Voice ⇒ Ditransitive!

Ipagaalis ng babae ng bigas ang bata
Will-take woman rice for-the child

VERB DET NOUN DET NOUN DET
Voice=Benf Case=Acc Case=Acc Case=Nom

Constituents vs. Dependencies
Plains Cree Transitive Clauses

- Nominal
- Bare NP
- Case
- Pre-verb
- Cross-ref on verb
- Agreement
- Post-verb
- Cross-ref on verb

Universal Dependencies
Direct-Inverse Voice in Plains Cree

Animacy hierarchy: 1st person > 3rd person
Direct-Inverse Voice in Plains Cree

Animacy hierarchy: 1st person > 3rd person
Should we set nsubj > obj?
Direct-Inverse Voice in Plains Cree

Constituents vs. Dependencies
The theme (not the recipient) is indirect object because it is not cross-referenced on the verb (it is inanimate, while the verb references an animate object).
Universal Dependencies

Ellipsis and Enhanced UD
1 Constituents vs. Dependencies

2 Universal Dependencies
   - A Tour through UD Syntax
   - Nonverbal Predicate and Copula
   - Core Arguments vs. Oblique Dependents
   - Ellipsis and Enhanced UD
Deleted Predicates in Coordination

Some treebanks would use an empty node to represent the second *went*. UD enhanced representation now allows empty nodes! But the basic representation sticks with the overt words.
UD V1: The remnant Relation

Kate went to Florida and Jane (went) to Europe
Kate went to Florida and Jane (went) to Europe
Perseus Treebanks: Chained Relations

Constituents vs. Dependencies
UD V2: The orphan Relation

Kate went to Florida and Jane (went) to Europe

Constituents vs. Dependencies

105/110
Kate wants to go to Florida and Jane (wants) (go) to Europe
Kate wants to go to Florida and Jane (wants) (go) to Europe.
Enhanced Dependencies: Coordination

Jane eats sweet apples and oranges

Constituents vs. Dependencies
A gdzie szukać szamponu, który myje?
And where to-look for-shampoo, that washes?
Basic Universal Dependencies: 104 (102) Languages and Growing

I.-E.: Armenian, Ancient Greek, Greek, Albanian, Breton, Irish, Manx, Scottish, Welsh, Afrikaans, Danish, Dutch, English, Faroese, German, Gothic, Icelandic, Norwegian, Swedish, Swiss German, Catalan, French, Galician, Italian, Latin, Old French, Portuguese, Romanian, Spanish, Belarusian, Bulgarian, Church Slavonic, Croatian, Czech, Old Russian, Polish, Russian, Serbian, Slovak, Slovenian, Ukrainian, Upper Sorbian, Latvian, Lithuanian, Kurmanji, Persian, Khunsari, Nayini, Soi, Urdu, Hindi, Bhojpuri, Marathi, Sanskrit

Dravidian: Tamil, Telugu

Uralic: Erzya, Estonian, Finnish, Hungarian, Karelain, Livvi, Komi Permyak+Zyrian, Moksha, Sámi North+Skolt

Turkic: Kazakh, Old Turkish, Turkish, Uyghur

Korean

Japanese

Sino-Tibetan: Cantonese, Classical Chinese, Chinese

Tai-Kadai: Thai

Aus.-As.: Vietnamese

Austronesian: Indonesian, Tagalog

Pama-Nyungan: Warlpiri

Chukotko-Kamchatkan: Chukchi

Arawakan: Apurinã

Tupian: Akuntsu, Mundurukú, Tupinambá, Mbyá Guarani

Af.-As.: Akkadian

Amharic, Arabic Standard+Levantine, Assyrian, Coptic, Hebrew, Maltese

Niger-Congo: Bambara, Wolof, Yoruba

Other: Basque, Sw. Sign, Naija

Constituents vs. Dependencies

109/110
Summary

• Constituent (phrase) trees ... context-free grammar
• Dependency trees (or graphs)
  • Nonprojective dependencies
• Universal Dependencies
  • Unified annotation for all languages
    • Language-specific extensions
  • Content words higher than function words ... better parallelism
• Clauses – nominals – modifier words
• Core arguments vs. oblique dependents

https://ufal.cz/courses/npfl094