Syntactic Analysis

Daniel Zeman

December 18, 2020
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
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))))

Universal Dependencies
Paul gave Peter two pears.
[#,0] (gave,2) ([Paul,1], [Peter,3], [pears,5] ([two,4])), [.,6])

Dependency Tree with Labels

# / AuxS
  
gave / Pred
  
Paul / Sb
  
Peter / Obj
  
pears / Obj
  
two / Atr
Paul gave Peter two pears

Dependency Tree with Labels

Root

Paul (nsubj)

Gave (iobj)

Peter (obj)

Two (nummod)
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**

![Phrase vs. Dependency Trees Diagram]

- **Phrase Tree:**
  - **S** (Sentence)
  - **NP:**
    - **N:** Paul
    - **V:** gave
    - **NP:**
      - **N:** Peter
      - **C:** two
      - **N:** pears

- **Dependency Tree:**
  - **root**
  - **nsubj:** Paul
  - **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.”

![Diagram of constituents and dependencies for Czech example]
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!

\[
\text{nepodařilo} / \text{Pred} \\
\text{se} / \text{AuxT} \quad \text{otevřít} / \text{Sb} \\
\text{soubor} / \text{Obj}
\]
Nonprojectivity Can Be Handled by a Dependency Tree!

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

16/110
Constituents vs. Dependencies
Constituents vs. Dependencies
My daughter bought some bread and cheese.
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
Universal Dependencies

- 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.
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

Constituents vs. Dependencies
Same Thing Same Way

Constituents vs. Dependencies
Same Thing Same Way

George killed the dragon

Mharaigh Seoirse an dragan

Jorge mató al dragón
George killed the dragon
Mharaigh Seoirse an dragan
Jorge mató a el dragón
George killed the dragon

Mharaigh Seoirse an dragan

Jorge mató a el dragón

Draka zabil Jiří
Same Meaning ≠ Same Construction!

He killed the dragon

PRON VERB DET NOUN

Constituents vs. Dependencies
Same Meaning ≠ Same Construction!

He killed the dragon

PRON VERB DET NOUN

The dragon was killed by him

DET NOUN AUX VERB ADP PRON

Constituents vs. Dependencies

24/110
Same Meaning ≠ Same Construction!

He killed the dragon

The dragon was killed by him

His killing of the dragon

Constituents vs. Dependencies
Same Meaning $\neq$ Same Construction!

- **He killed the dragon** (PRON VERB DET NOUN)
  - nsubj: He
  - obj: killed
  - det: the
  - nmod: DET
  - nmod: poss: His
  - case: killing

- **The dragon was killed by him** (DET NOUN AUX VERB ADP PRON)
  - nsubj:pass: The dragon
  - aux:pass: was
  - obl:agent: by
  - case: him

- **His killing of the dragon** (PRON NOUN ADP DET NOUN)
  - nmod:poss: His
  - case: killing

- **The dragon that was killed** (DET NOUN PRON AUX VERB)
  - nsubj:pass: The dragon
  - aux:pass: that was
  - acl:relcl: killed
राजा विष्णुशर्मानम् आहूया प्रोवाच
rājā viṣṇuśarmāṇam āhūya provāca

king Vishnusharma having-summoned said

NOUN PROPN VERB VERB

VerbForm=Conv VerbForm=Fin

Constituents vs. Dependencies
राजा विष्णुशरमणम् आहूया प्रोवाच
rājā viṣṇuśarmāṇam āhūya provāca

king Vishnusharma having-summoned said

NOUN PROPN VERB VERB
VerbForm=Conv VerbForm=Fin

Constituents vs. Dependencies
Universal Dependencies
A Tour through UD Syntax
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
- Content words are related by dependency relations
Content words are related by dependency relations

Function words attach to closest content words
Syntax

- 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.

Кучето се преследваше от котката.
The dog was chased by the cat.

Кучето се преследваше от котката.
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>nsubj</td>
<td>csubj</td>
<td>advmod</td>
<td>aux</td>
<td></td>
</tr>
<tr>
<td>expl</td>
<td>advcl</td>
<td>discourse</td>
<td>cop</td>
<td></td>
</tr>
<tr>
<td>expl</td>
<td>advcl</td>
<td>mark</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Non-Core</th>
<th>Nominal</th>
</tr>
</thead>
<tbody>
<tr>
<td>obl</td>
<td>expl</td>
</tr>
</tbody>
</table>

## Dependents of Nominals

<table>
<thead>
<tr>
<th>Nominal</th>
<th>Clausal</th>
<th>Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>nmod</td>
<td>acl</td>
<td>det</td>
</tr>
<tr>
<td>appos</td>
<td></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, as well as, ad hoc</em></td>
</tr>
<tr>
<td>flat</td>
<td><em>president Havel, New York, four thousand</em></td>
</tr>
<tr>
<td>compound</td>
<td><em>phone book, dress up</em></td>
</tr>
<tr>
<td>goeswith</td>
<td><em>not with standing, 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
### Other Relations

<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

<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:pri</td>
<td>Verb particle (dress up)</td>
</tr>
<tr>
<td>nmod:poss</td>
<td>Possessive nominal (Mary’s book)</td>
</tr>
<tr>
<td>obl:agent</td>
<td>Agent in passive (saved by the bell)</td>
</tr>
<tr>
<td>cc:preconj</td>
<td>Preconjuncton (both … and)</td>
</tr>
<tr>
<td>det:predet</td>
<td>Predeterminer (all 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:
  - Ivan is the best dancer.

- Some languages use a copula pronoun:
  - Ivan – to najlepszy tancerz.
  - Ivan – it best dancer.
Some languages use a copula verb:

- Ivan is the best dancer.

Some languages omit the copula:

- Иван лучший танцор.
Some languages use a copula verb:

Some languages use it only in some tenses:
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.
A clause can be the subject:

But it cannot be annotated as the nonverbal predicate:
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></td>
<td></td>
<td>discourse</td>
<td>cop</td>
</tr>
<tr>
<td>Non-Core</td>
<td>obl</td>
<td>advcl</td>
<td>mark</td>
<td></td>
</tr>
<tr>
<td></td>
<td>vocative</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>dislocated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></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></td>
<td>det</td>
</tr>
<tr>
<td>appos</td>
<td></td>
<td>amod</td>
<td>case</td>
</tr>
<tr>
<td></td>
<td></td>
<td>nummod</td>
<td></td>
</tr>
</tbody>
</table>
Information Packaging

Constituents vs. Dependencies
Information Packaging

Constituents vs. Dependencies
He loaded the wagon with hay.
Information Packaging

Constituents vs. Dependencies
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.)
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 ($\text{nsubj}$, $\text{obj}$, $\text{iobj}$)
  - Prepositional phrase $\Rightarrow$ oblique argument or adjunct ($\text{obl}$)

- 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…
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 $\Rightarrow$ 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...
  - 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
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 \( A \) becomes \( nsubj \)
  - function \( P \) becomes \( obj \)
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 \(\Rightarrow\) nsubj
  - function P \(\Rightarrow\) obj
Transitive Predicates in English

John kills Mary (primary transitive)

PROPN VERB PROPN

Constituents vs. Dependencies
Transitive Predicates in English

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

Constituents vs. Dependencies
Transitive Predicates in English

John *kills* Mary (primary transitive)

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

Constituents vs. Dependencies
Passivization in English

Constituents vs. Dependencies
Subject Control in English

Constituents vs. Dependencies

Universal Dependencies

PROPN VERB PART VERB PROPN

PROPN VERB PART VERB PROPN
Object Control in English

Constituents vs. Dependencies

62/110
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.*

![Diagram showing constituents vs. dependencies]
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
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

\[ \Rightarrow \text{function } S \]
- Regardless of semantic role:
  - \textit{John runs.}
  - \textit{John sleeps.}
  - \textit{John falls.}

- Then define:
  - function \( S \Rightarrow \text{nsubj} \)
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
```

- 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
  - VERB
    - Case=Nom
    - Voice=Act(Pass)
    - (ADP)
    - Case=Acc
      - bare NP
        - pre-verb
          - cross-ref on verb
    - declarative clause
      - agreement
      - post-verb
        - nomina

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

Pedro le dio un libro a Isabel

PROPN PRON VERB DET NOUN ADP PROPN

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
Czech Adjunct Exceptions

Pracuje celý týden
He-works whole week

Constituents vs. Dependencies
Czech Ditransitive Clauses

Constituents vs. Dependencies
Dative: Recipient vs. Beneficiary

Constituents vs. Dependencies
Monotransitive with Dative?

Zuzka pomohla Martinovi úkolem

PROPN VERB PROPN ADP NOUN
Case=Nom Case=Dat Gender=Neut Number=Sing Case=Ins

nsubj obl:arg case

Martinovi bylo pomoženo s úkolem homework

PROPN AUX ADJ ADP NOUN
Case=Dat Gender=Neut Number=Sing Case=Ins

obj? aux:pass obl:arg case
Monotransitive with Genitive?

Journalists must observe principles of objectivity.

Musí být dbáno zásad objektivity.
Monotransitive with Instrumental?

```
Karel
hýbal
nábytkem
Karel
moved
furniture
PROPN
VERB
NOUN
Case=Nom
Case=Ins

bylo
hýbáno
Furniture
was
moved
NOUN
AUX
ADJ
Case=Ins
Gender=Neut
Number=Sing
```

Constituents vs. Dependencies
Monotransitive with Preposition?

Constituents vs. Dependencies
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”

- UD Czech 2.1–2.5: bare NP > PP
  - Nom, Acc, Gen, Dat, Ins

- UD Czech 2.6 (May 2020):
  - Nom, Acc > Gen, Dat, Ins, ADP + adjuncts

- No ditransitives in Czech!
- (Exception: učit “to teach” takes two Acc.)
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

Constituents vs. Dependencies
Basque Transitive Clauses

Constituents vs. Dependencies
Basque Intransitive Clauses

Constituents vs. Dependencies
Basque Ditransitive Clauses

Constituents vs. Dependencies
Basque Causative Applied to Dative Subject

Constituents vs. Dependencies
Yidiŋ Transitive Clauses

nominal → VERB → nominal

Case=Erg
Case=Nom

bare NP

nsubj

root

obj

Case=Abs
Case=Acc

case=Acc

bare NP
“I, (who) was slapped by the woman, laughed”
“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.
"I, (who) was slapping the woman, laughed"

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

Constituents vs. Dependencies
Magaalis ang babae ng sako
VERB DET NOUN DET NOUN DET NOUN
Voice=Act? Case=Nom Case=Acc Case=Loc

Aalisin ng babae ang bigas sa sako
VERB DET NOUN DET NOUN DET NOUN
Voice=Pass? Case=Acc Case=Nom Case=Loc

Constituents vs. Dependencies
Tagalog Locative Voice ⇒ Ditransitive!

Aalisan
Will-take
VERB
Voice=Locf

ng
DET
Case=Acc

babae
woman
NOUN
Case=Acc

ng
DET
Case=Acc

bigas
rice
NOUN
Case=Nom

ang
from-the
DET

sako
sack
NOUN

obj:agent

nsubj:loc

det

det

det

obj:patient

Case=Acc

Case=Nom
Tagalog Benefactive Voice ⇒ Ditransitive!

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

CONSTITUENTS:
- VERB: Ipagaalis
- DET: ng
- NOUN: babae
- DET: ng
- NOUN: bigas
- DET: ang
- NOUN: bata

DEPENDENCIES:
- root
- nsubj:ben
- det
- obj:agent
- det
- obj:patient
- det

Voice=Benf
Case=Acc
Case=Acc
Case=Nom
Plains Cree Transitive Clauses

Constituents vs. Dependencies
Direct-Inverse Voice in Plains Cree

Animacy hierarchy: 1st person > 3rd person

Niwįcihānānak
We-help-them
VERB
Voice=Dir

Niwįcihikonānak
They-help-us
VERB
Voice=Inv
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).
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
Deleted Predicates in Coordination

Kate went to Florida and Jane (went) to Europe

- 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.
Kate went to Florida and Jane (went) to Europe
Kate went to Florida and Jane (went) to Europe
Kate went to Florida and Jane (went) to Europe
UD V2: The orphan Relation

Constituents vs. Dependencies
Enhanced Dependencies: Gapping

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, Karelian, 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 Guaraní
  - **Af.-As.**:
    - Akkadian, Arabic Standard+Levantine, Assyrian, Coptic, Hebrew, Maltese
  - **Niger-Congo**:
    - Bambara, Wolof, Yoruba
  - **Other**:
    - Basque, Sw. Sign, Naija

*Constituents vs. Dependencies*
Syntactic Analysis

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/nplf1094