Syntactic Analysis

Daniel Zeman

March 8, 2023
<table>
<thead>
<tr>
<th>ID</th>
<th>FORM</th>
<th>LEMMA</th>
<th>POS</th>
<th>FEATS</th>
<th>HEAD</th>
<th>DEPREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>They</td>
<td>they</td>
<td>PRON</td>
<td>Case...</td>
<td>2</td>
<td>nsubj</td>
</tr>
<tr>
<td>2</td>
<td>buy</td>
<td>buy</td>
<td>VERB</td>
<td>Mood...</td>
<td>0</td>
<td>root</td>
</tr>
<tr>
<td>3</td>
<td>and</td>
<td>and</td>
<td>CCONJ</td>
<td>_</td>
<td>4</td>
<td>cc</td>
</tr>
<tr>
<td>4</td>
<td>sell</td>
<td>sell</td>
<td>VERB</td>
<td>Mood...</td>
<td>2</td>
<td>conj</td>
</tr>
<tr>
<td>5</td>
<td>books</td>
<td>book</td>
<td>NOUN</td>
<td>Number...</td>
<td>2</td>
<td>obj</td>
</tr>
<tr>
<td>6</td>
<td>.</td>
<td>.</td>
<td>PUNCT</td>
<td>_</td>
<td>2</td>
<td>punct</td>
</tr>
</tbody>
</table>
Dependency Tree

They buy and sell books.

Syntactic Analysis
Constituents vs. Dependencies Phrases, Their Types and Their Heads Dependency Parsing
They buy and sell books.
### Syntactic Annotation

<table>
<thead>
<tr>
<th>ID</th>
<th>FORM</th>
<th>LEMMA</th>
<th>POS</th>
<th>FEATS</th>
<th>HEAD</th>
<th>DEPREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Kupují</td>
<td>kupovat</td>
<td>VERB</td>
<td>Mood...</td>
<td>0</td>
<td>root</td>
</tr>
<tr>
<td>2</td>
<td>a</td>
<td>a</td>
<td>CCONJ</td>
<td>_</td>
<td>3</td>
<td>cc</td>
</tr>
<tr>
<td>3</td>
<td>prodávají</td>
<td>prodávat</td>
<td>VERB</td>
<td>Mood...</td>
<td>1</td>
<td>conj</td>
</tr>
<tr>
<td>4</td>
<td>knihy</td>
<td>kniha</td>
<td>NOUN</td>
<td>Case...</td>
<td>1</td>
<td>obj</td>
</tr>
<tr>
<td>5</td>
<td>.</td>
<td>.</td>
<td>PUNCT</td>
<td>_</td>
<td>1</td>
<td>punct</td>
</tr>
</tbody>
</table>
### Syntactic Annotation

<table>
<thead>
<tr>
<th>ID</th>
<th>FORM</th>
<th>LEMMA</th>
<th>POS</th>
<th>FEATS</th>
<th>HEAD</th>
<th>DEPREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Kupují</td>
<td>kupovat</td>
<td>VERB</td>
<td>Mood…</td>
<td>0</td>
<td>root</td>
</tr>
<tr>
<td>2</td>
<td>a</td>
<td>a</td>
<td>CCONJ</td>
<td>_</td>
<td>3</td>
<td>cc</td>
</tr>
<tr>
<td>3</td>
<td>prodávají</td>
<td>prodávat</td>
<td>VERB</td>
<td>Mood…</td>
<td>1</td>
<td>conj</td>
</tr>
<tr>
<td>4</td>
<td>knihy</td>
<td>kniha</td>
<td>NOUN</td>
<td>Case…</td>
<td>1</td>
<td>obj</td>
</tr>
<tr>
<td>5</td>
<td>.</td>
<td>.</td>
<td>PUNCT</td>
<td>_</td>
<td>1</td>
<td>punct</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>FORM</th>
<th>LEMMA</th>
<th>XPOS</th>
<th>HEAD</th>
<th>DEPREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Kupují</td>
<td>kupovat</td>
<td>VB-P-----3P-AA---</td>
<td>2</td>
<td>Pred_Co</td>
</tr>
<tr>
<td>2</td>
<td>a</td>
<td>a</td>
<td>J^-------------</td>
<td>0</td>
<td>Coord</td>
</tr>
<tr>
<td>3</td>
<td>prodávají</td>
<td>prodávat</td>
<td>VB-P-----3P-AA---</td>
<td>2</td>
<td>Pred_Co</td>
</tr>
<tr>
<td>4</td>
<td>knihy</td>
<td>kniha</td>
<td>NNFP4-----A----</td>
<td>2</td>
<td>Obj</td>
</tr>
<tr>
<td>5</td>
<td>.</td>
<td>.</td>
<td>Z:-------------</td>
<td>0</td>
<td>AuxK</td>
</tr>
</tbody>
</table>
Kupují a prodávají knihy.
Constituents vs. Dependencies
What Is (Surface) Syntax?

• Relations between sentence parts
• Sentence part = token (word, number, punctuation)
  • Advantages:
    • Token is easily recognizable
    • Unit of previous (morphological) level of processing
    • We do not restore elided (missing) constituents at this level
  • Drawbacks:
    • We must define relations for function words and punctuation

• Purpose:
  • Understand how meaning of words combines into meaning of sentence
  • Some theories also want to show how the sentence was generated
• Different shapes in different theories
• Typically a hierarchical structure – tree

  • Phrasal (constituent) tree, parse tree

  • Dependency tree
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])

Syntactic Analysis

Phrases, Their Types and Their Heads  Dependency Parsing
Paul gave Peter two pears
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

**Syntactic Analysis**

**Constituents vs. Dependencies**

**Phrases, Their Types and Their Heads**

**Dependency Parsing**

---

Phrasal vs. Dependency Trees

S

NP → VP

N → V → NP → NP

Paul gave N → C → N → Peter two pears

Paul gave Peter two pears
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

Syntactic Analysis

Constituents vs. Dependencies

Phrases, Their Types and Their Heads  Dependency Parsing
Phrases, Their Types and Their Heads
Phrase Replaceability

• A phrase can be replaced by another phrase of the same type
• The sentence stays grammatical
• Specifically, a phrase can be replaced by its head
  • This is related to the generation of the sentence

• Phrases $x, y, z$ can be immediate constituents of a larger phrase $f$ only if they are related to each other. This is however a matter of the particular phrase structure grammar.

• Example sentence: *This is the man that I talked about.*
  • The part *man that I* is not a whole noun phrase.
  • Cannot replace it by another noun phrase, e.g., *man:*
  • *This is the man talked about.*
• Sequence of immediate constituents (words or phrases)
• May be discontinuous in some languages
• Phrase types by their main word—head
  • Noun phrase: the new **book** of my grandpa
  • Adjectival phrase: **brand** new
  • Adverbial phrase: very **well**
  • Prepositional phrase: **in the classroom** (if preposition is considered the head—somewhat controversial (cannot replace whole phrase by preposition))
  • Verb phrase: to **catch** a **ball**
A noun or a (substantive) pronoun is the head.

- water
- the book
- new ideas
- two millions of inhabitants
- one small village
- the greatest price movement in one year since the World War II
- operating system that, regardless of all efforts by our admin, crashes just too often
- he
- whoever
An adjective or a determiner (attributive pronoun) is the head.

Simple adjectives are very frequent, complex ADJPs are rare.

- *old*
- *very old*
- *really very old*
- *five times older than the oldest elephant in our zoo*
- *sure that he will arrive first*
Pronouns / Determiners

- (Substantive) pronouns: similar behavior as nouns
  - Personal pronouns \((I, you, they, oneself)\)
  - Some demonstrative, interrogative, relative and negative pronouns \((who, what, somebody, something, everything, nothing)\)

- Attributive pronouns (determiners): similar behavior as adjectives
  - Possessive pronouns in some languages \((my, your, his, whose)\)
  - Articles \((the, a, an)\)
  - Attributively used demonstrative, interrogative, relative and negative pronouns \((which, some, every, no)\)
Numerals and Quantified Noun Phrases

• Slavic languages: not always clear what should be the head: the quantifier (number), or the counted noun?
  • Numeral inherits gender from counted noun.
  • Counted noun gets grammatical number from numeral (or in accord with it).
    • jeden muž “one man”, jedna žena “one woman”, jedno dítě “one child”
    • dva muži “two men”, dvě ženy “two women”, dvě děti “two children”
  • Numeral may govern the case of the counted noun.
    • pět mužů “five men” – noun in genitive, numeral in nominative, accusative, or vocative
  • Or both numeral and counted noun have case required by preposition or verb.
    • pěti ženami “five women” – instrumental case
• An adverb is the head.
• Simple adverbs are very frequent, complex ADVPs are rare.
  • *quickly*
  • *much more*
  • *how*
  • *louder than you can imagine*
  • *yesterday*
Prepositional (Postpositional) Phrase

• Many theories: preposition is the head (it determines the case of the rest of the phrase)
• But we cannot replace the phrase by the preposition alone! (Nor can we replace it by the noun without the preposition.)
• PPs often have functions similar to adverbial phrases or noun phrases.
• Preposition in one language may correspond to case morphology in another language.
  • \textit{in} the city center
  • \textit{in} January
  • \textit{in} God
  • \textit{around} five o’clock
  • \textit{to} a better future
  • \textit{up to} a situation where neither of them could back out
  • \textit{with respect to} his nonage
Ambiguity in Attachment of Prepositional Phrases

- Classic English example:
  - *I saw the man with a telescope.*
    1. “Viděl jsem muže dalekohledem.”
    2. “Viděl jsem muže s dalekohledem.”
Přišel ten pán se sousedem odnaproti. “The gentleman came with his neighbor from across the street.”
(Odnaproti is an adverb arisen as a frozen PP: od + naproti.)
Přišel ten pán se sousedem odnaproti. “The gentleman came with his neighbor from across the street.”

(Odnaproti is an adverb arisen as a frozen PP: od + naproti.)
Přišel ten pán se sousedem odnaproti. “The gentleman came with his neighbor from across the street.”

(*Odnaproti* is an adverb arisen as a frozen PP: *od* + *naproti*.)
Ambiguity in Attachment of Prepositional Phrases: Czech Example

Přišel ten pán se sousedem odnaproti. “The gentleman came with his neighbor from across the street.”

(Odnaproti is an adverb arisen as a frozen PP: od + naproti.)
• V letech 1991 – 1993 jsem absolvovala kurzy řízení a marketingu na Collège Bart v kanadském Québecu.

• “In years 1991 – 1993 I attended classes of management and marketing at Collège Bart in Canadian Québec.”

(A Czech sentence from the Prague Dependency Treebank.)
Prepositional Phrase Attachment Ambiguity: Corpus Example

• In years 1991 – 1993 I attended classes of management and marketing at Collège Bart in Canadian Québec.

  • attended at Collège Bart
  • classes at Collège Bart
  • management and marketing at Collège Bart
  • marketing at Collège Bart
  • Collège Bart in Québec
  • marketing in Québec
  • management and marketing in Québec
  • classes in Québec
  • attended in Québec
In years 1991 – 1993 I attended classes of management and marketing at Collège Bart in Canadian Québec.

- attended (classes (of (management and marketing))) (at Collège Bart)
- attended (classes (of (management and marketing))) (at Collège Bart))
- attended (classes (of (management and marketing)) (at Collège Bart)))
- attended (classes (of (management and (marketing (at Collège Bart)))))
- … ((at Collège Bart) (in Québec))
  - Is Collège Bart in Québec or Québec in Collège Bart?
Prepositional Phrase Attachment Ambiguity: News Example

- říjnové jednání OSN o klimatických změnách v Kodani (Události ČT, 27.2.2009)
- “October UNO summit about climatic changes in Copenhagen” (Czech TV news, 2009-02-27)

- Question: Were there climatic changes in Copenhagen?
The repertory depends on the rules for analytical verb forms and varies greatly across languages. 

- *it rains*
- *he could at all sight Mr. President*
- *why we got wet so much*
- *Go!*
- *he has been transported to the hospital on Sunday*
- *it began to rain*
- *the law prohibits smoking in this room*
- *give Mary the beads that we brought from the vacation in Morocco*
- *the file could not be opened*
• One predicate together with its arguments and modifiers, e.g.:
  • *John loves Mary.*
  • *... that you are right.*

• Not recursive $\Rightarrow$ not necessarily the same as a verb phrase (VP).
  • Nested phrases are part of the superordinate phrase.
  • Nested clauses are not parts of the main (‘matrix’) clause.
Coordination

- There is no real head ⇒ difficult to capture in dependency trees.
- The coordinate phrases (conjuncts) are usually of the same type.
  - chickens, hens, rabbits, cats and dogs
  - new or even newer
  - quickly and finely
  - he came to the conclusion that there is no point in hiding any more, so we might hear him here today
  - in the house or outside
  - to and from Prague
  - either now or later
  - not only on Monday and on Wednesday but also tomorrow or the day after tomorrow
  - veni, vidi, vici
Ellipsis

- A phrase omitted from the (surface form of the) sentence although it is present in the underlying meaning (deep structure).
- Often in dialogues: the elided information is known from context.
  - *Whom did you see there? — Peter.* (missing verb)
- Also often in coordination:
  - *Czech and German researchers discussed*... It probably means *Czech researchers and German researchers discussed*. Unlikely that each researcher was Czech and German at the same time.
  - *The Penguins are leading 4:0, while the Colorado Avalanches only 3:2.* (missing verb in the second part)
- Systemic elision of subject pronouns in pro-drop languages (it is marked on the verb).
  - *Sedím. “(I) sit.”*
Dependency Parsing
Dependency Parsing

- Automatically assign a dependency tree to a sentence.
- Machine-learning: manually annotated “gold standard” data needed!

- **Chart** (Eisner, CKY)
  - $O(n^3)$
  - Produces only projective parses
    (if $x$ directly depends on $y$, all words between $x$ and $y$ transitively depend on $y$)
Dependency Parsing

- Automatically assign a dependency tree to a sentence.
- Machine-learning: manually annotated “gold standard” data needed!

- **Chart** (Eisner, CKY)
  - \( O(n^3) \)
  - Produces only projective parses
    (if \( x \) directly depends on \( y \), all words between \( x \) and \( y \) transitively depend on \( y \))

- **Transition-based** (shift-reduce)
  - \( O(n) \) (fast!)
  - Can be extended to capture nonprojectivity
Dependency Parsing

• Automatically assign a dependency tree to a sentence.
• Machine-learning: manually annotated “gold standard” data needed!

• **Chart** (Eisner, CKY)
  • \(O(n^3)\)
  • Produces only projective parses
    (if \(x\) directly depends on \(y\), all words between \(x\) and \(y\) transitively depend on \(y\))

• **Transition-based** (shift-reduce)
  • \(O(n)\) (fast!)
  • Can be extended to capture nonprojectivity

• **Graph-based** (MST)
  • \(O(n^2)\)
  • Can produce projective and nonprojective parses
Transition-Based Parsers: Malt

- Nivre et al., *Natural Language Engineering* 2007
- [http://maltparser.org/](http://maltparser.org/)
- Based on *transitions* from one configuration to another
- Configuration:
  - Input buffer (words of the sentence, left-to-right)
  - Stack
  - Output tree (words, edges, labels)
- Transitions:
  - *Shift*: move word from buffer to stack
  - *Larc*: connect two topmost stack words (higher is parent)
  - *Rarc*: connect two topmost stack words (lower is parent)
Malt Parser

- Driven by oracle
  - Looks at current configuration
  - Selects next transition
• Driven by oracle
  • Looks at current configuration
  • Selects next transition
• Training: decompose the training tree to a sequence of transitions
  • Sometimes more than one possibility
    • Various learning strategies: e.g. create dependencies eagerly, as soon as possible
Malt Parser

- Driven by **oracle**
  - Looks at current configuration
  - Selects next transition
- **Training**: decompose the training tree to a sequence of transitions
  - Sometimes more than one possibility
    - Various learning strategies: e.g. create dependencies eagerly, as soon as possible
- The oracle learns based on the features of the configuration
  - E.g. word, lemma, POS, case, number...
    - $n^{th}$ word from the top of the stack
    - $k^{th}$ word remaining in the buffer
    - particular node in output tree part created so far
• Machine learning responsible for training, here the Support Vector Machines (SVM)
  • Classifier. Input vectors: values of all features of the current configuration
  • In addition, during training there is the output value, i.e. action identifier (shift / larc / rarc)
  • The trained oracle (SVM) tells the output value during parsing
• Training on the whole PDT may take weeks!
  • Complexity $O(n^2)$ where $n$ is number of training examples
  • Over 3 million training examples can be extracted from PDT
• Parsing comparatively faster (~ 1 sentence / second) and can be parallelized
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT</td>
<td>Pavel dal Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT</td>
<td>Pavel dal Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

SHIFT

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
### Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT Pavel</td>
<td>dal Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky .  
Pavel gave Petr two pears .
## Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT Pavel</td>
<td>dal Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

**SHIFT**

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT Pavel dal</td>
<td>Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT Pavel dal</td>
<td>Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

**LARC**

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td>Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td>Petrovi dvě hrušky .</td>
</tr>
</tbody>
</table>

SHIFT

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal Petrovi</td>
<td>dvě hrušky .</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal Petrovi</td>
<td>dvě hrušky .</td>
</tr>
</tbody>
</table>

RARC

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td>dvě hrušky .</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td>dvě hrušky .</td>
</tr>
</tbody>
</table>

```
SHIFT
```

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .

Syntactic Analysis
Constituents vs. Dependencies Phrases, Their Types and Their Heads
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal dvě</td>
<td>hrušky</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky.
Pavel gave Petr two pears.
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT</td>
<td>dal dvě</td>
</tr>
<tr>
<td>hrušky</td>
<td></td>
</tr>
</tbody>
</table>

**SHIFT**

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
### Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal dvě hrušky</td>
<td>.</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky. Pavel gave Petr two pears.
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT</td>
<td>dal</td>
</tr>
<tr>
<td>dvě</td>
<td>hrušky</td>
</tr>
<tr>
<td>.</td>
<td></td>
</tr>
</tbody>
</table>

LARC

Pavel dal Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal hrušky</td>
<td>.</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky.
Pavel gave Petr two pears.
Example of Malt Parsing

STACK | BUFFER
--- | ---
ROOT *dal hrušky* | .

RARC

Pavel *dal* Petrovi dvě hrušky .
Pavel gave Petr two pears .
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td>.</td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky.
Pavel gave Petr two pears.
Example of Malt Parsing

Stack:

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td></td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky.
Pavel gave Petr two pears.

SHIFT
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT</td>
<td>dal</td>
</tr>
</tbody>
</table>

Pavel gave Petr two pears.

Constituents vs. Dependencies
Phrases, Their Types and Their Heads
Example of Malt Parsing

**STACK** | **BUFFER**
---|---
**ROOT** `dal` .

**RARC**

Pavel `dal` Pavel gave Petr `dvě hrušky`.
Pavel gave Petr `dvě hrušky`.

**Constituents vs. Dependencies**
**Phrases, Their Types and Their Heads**
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td></td>
</tr>
</tbody>
</table>

Pavel gave Petr two pears.
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT dal</td>
<td></td>
</tr>
</tbody>
</table>

RARC

Pavel dal Petroví dvě hrušky.
Pavel gave Petr two pears.
Example of Malt Parsing

<table>
<thead>
<tr>
<th>STACK</th>
<th>BUFFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT</td>
<td></td>
</tr>
</tbody>
</table>

Pavel dal Petrovi dvě hrušky.
Pavel gave Petr two pears.
So Far Only Projective Trees

- It can be proven that the above transition system is correct.
  - resulting graph is always a tree (connected, cycle-free)

- There are extensions that can produce non-projective trees
- Non-projective constructions occur in natural languages but they are rare
So Far Only Projective Trees

- It can be proven that the above transition system is:
  - **correct**
    - resulting graph is always a tree (connected, cycle-free)
  - **complete** for the set of projective trees
    - every projective tree can be expressed as a sequence of transitions

- There are extensions that can produce non-projective trees
- Non-projective constructions occur in natural languages but they are rare