Mining Unstructured Data
6. Constituent parsing
Outline

1. Syntactic parsing
   - Goal and motivation

2. Trees and Grammars

3. Constituent Parsing
   - Background
   - Chart-based methods
   - CKY Algorithm
Outline

1. Syntactic parsing
   - Goal and motivation

2. Trees and Grammars

3. Constituent Parsing
   - Background
   - Chart-based methods
   - CKY Algorithm
Syntax studies the combination of words in a sentence.  
Syntactic parsing provides information of the combination of words in a sentence (the syntactic structure).  
Syntactic information is relevant for many NLP applications:  
  - Authorship recognition  
  - Grammar checking  
    - Ex: 3th-Singular-noun + basic-verb $\rightarrow$ error  
  - Machine Translation  
    - Ex: [es] NN+JJ $\rightarrow$ [en] JJ+NN  
  - Information Extraction  
    - Ex: $X - [subj] \rightarrow visited \leftarrow [dobj] - Y \rightarrow visit(X,Y)$  
  - ...  

**Goal:** find the syntactic structure associated to a sentence.
Outline

1. Syntactic parsing
   - Goal and motivation

2. Trees and Grammars

3. Constituent Parsing
   - Background
   - Chart-based methods
   - CKY Algorithm
Mining Unstructured Data

Syntactic parsing
Trees and Grammars
Constituent Parsing

A Syntactic Tree

```
S
  NP
    PRP
    They
  VP
    VBD
    solved
    NP
    DT
    the
    NN
    problem
    PP
    IN
    with
    NNS
    statistics
```
Another Syntactic Tree

```
S
  NP
    PRP They
    VBD solved
  VP
    NP
      DT the
      NN problem
      PP
        IN with
        NNS statistics
```
Dependency Trees

They solved the problem with statistics.
They solved the problem with statistics.
Influential members of the House Ways and Means Committee introduced legislation that would restrict how the new savings-and-loan bailout agency can raise capital, creating another potential obstacle to the government's sale of sick thrifts.
Theories of Syntactic Structure

Constituent Trees

- Main element: constituents (or phrases, or bracketings)
- Constituents = abstract linguistic units
- Results in nested trees

Dependency Trees

- Main element: dependency
- Focus on relations between words
- Handles free word order nicely.
Context Free Grammars (CFGs)

A context-free grammar is defined as a tuple $G = \langle N, \Sigma, R, S \rangle$ where:

- $N$ is a set of non-terminal symbols
- $S \in N$ is a distinguished start symbol
- $\Sigma$ is a set of terminal symbols
- $R$ is a set of rules of the form $X \rightarrow Y_1Y_2\ldots Y_n$ where $n \geq 0$, $X \in N$, $Y_i \in N \cup \Sigma$
Context Free Grammars, Example

\[ N = \{ S, \text{VP, NP, PP, DT, Vi, Vt, NN, IN} \} \]
\[ S = \{ S \} \]
\[ \Sigma = \{ \text{sleeps, saw, man, woman, telescope, the, with, in} \} \]
\[ R = \{ \begin{align*}
S & \rightarrow \text{NP VP} & \text{Vi} & \rightarrow \text{sleeps} \\
\text{NP} & \rightarrow \text{DT NN} & \text{Vt} & \rightarrow \text{saw} \\
\text{NP} & \rightarrow \text{NP PP} & \text{NN} & \rightarrow \text{man} \\
\text{PP} & \rightarrow \text{IN NP} & \text{NN} & \rightarrow \text{woman} \\
\text{VP} & \rightarrow \text{Vi} & \text{NN} & \rightarrow \text{telescope} \\
\text{VP} & \rightarrow \text{Vt NP} & \text{DT} & \rightarrow \text{the} \\
\text{VP} & \rightarrow \text{VP PP} & \text{IN} & \rightarrow \text{with} \\
\text{IN} & \rightarrow \text{in} 
\end{align*} \} \]

\[1 \text{S=sentence, VP=verb phrase, NP=noun phrase, PP=prepositional phrase, DT=determiner, Vi=intransitive verb, Vt=transitive verb, NN=noun, IN=preposition} \]
Properties of CFGs

- A CFG defines a set of possible derivations (i.e. unique trees).
- A sequence of terminals $s \in \Sigma^*$ is generated by the CFG (or recognized by it, or belongs to the language defined by it) if there is at least a derivation that produces $s$.
- Some sequences of terminals generated by the CFG may have more than one derivation (ambiguity).
Ambiguity

- Mary used a telescope to see a man who was in the mountain
- Mary saw a man who was in the mountain and carried a telescope
- Mary was in the mountain and used a telescope to see a man
- Mary was in the mountain that has a telescope and saw a man
- Mary saw a man who was in the mountain that has a telescope
- Mary was in the mountain and saw a man carrying a telescope
She announced a program to promote safety in trucks and vans.

- She announced a program aimed to make trucks and vans safer.
- She used trucks and vans to announce a program aimed to promote safety.
- She announced a program aimed to make trucks safer. She also announced vans.
- She used trucks to announce a program aimed to promote safety. She also announced vans.
- She announced a program. She did so in order to promote safety in trucks and vans.
- She used trucks and vans to announce a program. She did so in order to promote safety.
- ...
Ambiguity

Some trees are more likely than others...
Ambiguity

Some trees are more likely than others...

Can we model that?
Context Free Grammar (CFGs)

A context-free grammar is defined as a tuple $G = \langle N, \Sigma, R, S \rangle$ where:

- $N$ is a set of non-terminal symbols
- $S \in N$ is a distinguished start symbol
- $\Sigma$ is a set of terminal symbols
- $R$ is a set of rules of the form $X \rightarrow Y_1 Y_2 \ldots Y_n$ where $n \geq 0$, $X \in N$, $Y_i \in N \cup \Sigma$
Context Free Grammar (CFGs)

A context-free grammar is defined as a tuple $G = \langle N, \Sigma, R, S \rangle$ where:

- $N$ is a set of non-terminal symbols
- $S \in N$ is a distinguished start symbol
- $\Sigma$ is a set of terminal symbols
- $R$ is a set of rules of the form $X \rightarrow Y_1Y_2\ldots Y_n$ where $n \geq 0$, $X \in N$, $Y_i \in N \cup \Sigma$
Probabilistic Context Free Grammar (PCFGs)

A context-free grammar is defined as a tuple $G = \langle N, \Sigma, R, S \rangle$ where:
- $N$ is a set of non-terminal symbols
- $S \in N$ is a distinguished start symbol
- $\Sigma$ is a set of terminal symbols
- $R$ is a set of rules of the form $X \rightarrow Y_1Y_2\ldots Y_n$ where $n \geq 0$, $X \in N$, $Y_i \in N \cup \Sigma$
A **probabilistic** context-free grammar is defined as a tuple \( G = \langle N, \Sigma, R, S \rangle \) where:

- \( N \) is a set of non-terminal symbols
- \( S \in N \) is a distinguished start symbol
- \( \Sigma \) is a set of terminal symbols
- \( R \) is a set of rules of the form \( X \rightarrow Y_1Y_2\ldots Y_n \) where \( n \geq 0, \ X \in N, \ Y_i \in N \cup \Sigma \)
Probabilistic Context Free Grammar (PCFGs)

A probabilistic context-free grammar is defined as a tuple $G = \langle N, \Sigma, R, S, q \rangle$ where:

- $N$ is a set of non-terminal symbols
- $S \in N$ is a distinguished start symbol
- $\Sigma$ is a set of terminal symbols
- $R$ is a set of rules of the form $X \rightarrow Y_1Y_2\ldots Y_n$ where $n \geq 0$, $X \in N$, $Y_i \in N \cup \Sigma$
- $q$ is a set of non-negative parameters, one for each rule $X \rightarrow \alpha \in R$ such that, for any $X \in N$, $\sum (X \rightarrow \alpha) \in R q(X \rightarrow \alpha) = 1$
Probabilistic Context Free Grammar (PCFGs)

A probabilistic context-free grammar is defined as a tuple \( G = \langle N, \Sigma, R, S, q \rangle \) where:

- \( N \) is a set of non-terminal symbols
- \( S \in N \) is a distinguished start symbol
- \( \Sigma \) is a set of terminal symbols
- \( R \) is a set of rules of the form \( X \rightarrow Y_1Y_2\ldots Y_n \) where \( n \geq 0, \ X \in N, \ Y_i \in N \cup \Sigma \)
- \( q \) is a set of non-negative parameters, one for each rule \( X \rightarrow \alpha \in R \) such that, for any \( X \in N \),

\[
\sum_{(X \rightarrow \alpha) \in R} q(X \rightarrow \alpha) = 1
\]
Context Free Grammars, Example

\[ N = \{S, \text{VP}, \text{NP}, \text{PP}, \text{DT}, \text{Vi}, \text{Vt}, \text{NN}, \text{IN}\} \]

\[ S = \{S\} \]

\[ \Sigma = \{\text{sleeps}, \text{saw}, \text{man}, \text{woman}, \text{telescope}, \text{the}, \text{with}, \text{in}\} \]

\[ R = \begin{cases} 
\text{S} \to \text{NP} \ \text{VP} & \text{Vi} \to \text{sleeps} \\
\text{NP} \to \text{DT} \ \text{NN} & \text{Vt} \to \text{see} \\
\text{NP} \to \text{NP} \ \text{PP} & \text{NN} \to \text{man} \\
\text{PP} \to \text{IN} \ \text{NP} & \text{NN} \to \text{woman} \\
\text{VP} \to \text{Vi} & \text{NN} \to \text{telescope} \\
\text{VP} \to \text{Vt} \ \text{NP} & \text{DT} \to \text{the} \\
\text{VP} \to \text{VP} \ \text{PP} & \text{IN} \to \text{with} \\
\text{IN} \to \text{in} & \end{cases} \]

\[ 1 \text{S}=\text{sentence}, \text{VP}=\text{verb phrase}, \text{NP}=\text{noun phrase}, \text{PP}=\text{prepositional phrase}, \text{DT}=\text{determiner}, \text{Vi}=\text{intransitive verb}, \text{Vt}=\text{transitive verb}, \text{NN}=\text{noun}, \text{IN}=\text{preposition} \]
# Probabilistic Context Free Grammars, Example

\[ N = \{ S, VP, NP, PP, DT, Vi, Vt, NN, IN \} \]
\[ S = \{ S \} \]
\[ \Sigma = \{ \text{sleeps, saw, man, woman, telescope, the, with, in} \} \]

\[ R = \begin{cases} 
S \rightarrow NP \ VP & 1.0 \\
NP \rightarrow DT \ NN & 0.4 \\
NP \rightarrow NP \ PP & 0.6 \\
PP \rightarrow IN \ NP & 1.0 \\
VP \rightarrow Vi & 0.5 \\
VP \rightarrow Vt \ NP & 0.4 \\
VP \rightarrow VP \ PP & 0.1 \\
\end{cases} \begin{cases} 
Vi \rightarrow \text{sleeps} & 1.0 \\
Vt \rightarrow \text{saw} & 1.0 \\
NN \rightarrow \text{man} & 0.7 \\
NN \rightarrow \text{woman} & 0.2 \\
NN \rightarrow \text{telescope} & 0.1 \\
DT \rightarrow \text{the} & 1.0 \\
IN \rightarrow \text{with} & 0.5 \\
IN \rightarrow \text{in} & 0.5 \\
\end{cases} \]

---

\(^1\)S=sentence, VP=verb phrase, NP=noun phrase, PP=prepositional phrase, DT=determiner, Vi=intransitive verb, Vt=transitive verb, NN=noun, IN=preposition
Properties of PCFGs

- The probability of a parse tree $t \in \mathcal{T}_G$ is computed as:
  \[
p(t) = \prod_{r \in t} q(r)
  \]

- If there is more than one tree for a sentence, we can rank them by probability.

- The most likely tree for a sentence $s$ is:
  \[
  \arg \max_{t \in \mathcal{T}(s)} p(t)
  \]
Learning Treebank Grammars

- Read the grammar rules from a treebank

```
S
  NP  VP
    PRP VBD NP
      She heard DT NN
        the noise

S \rightarrow NP VP . 1
NP \rightarrow PRP 0.5
NP \rightarrow DT NN 0.5
VP \rightarrow VBD NP 1
PRP \rightarrow She 1
...
```

- Set rule weights by maximum likelihood

\[
q(\alpha \rightarrow \beta) = \frac{\text{Count}(\alpha \rightarrow \beta)}{\text{Count}(\alpha)}
\]

- Smoothing issues apply

- Having the appropriate CFG is critical to success
Outline

1. Syntactic parsing
   - Goal and motivation

2. Trees and Grammars

3. Constituent Parsing
   - Background
   - Chart-based methods
   - CKY Algorithm
Outline

1. Syntactic parsing
   - Goal and motivation

2. Trees and Grammars

3. Constituent Parsing
   - Background
     - Chart-based methods
     - CKY Algorithm
Possible goals of a parser:

Find all possible trees
Parsing Natural Language Sentences

Possible goals of a parser:

Find all possible trees, maybe ranked by probability
Parsing Natural Language Sentences

Possible goals of a parser:

- Find all possible trees, maybe ranked by probability or find the most likely tree.
Parsing Natural Language Sentences

Possible goals of a parser:

Find all possible trees, maybe ranked by probability or find the most likely tree.

Parsing performance depends on many aspects:
Parsing Natural Language Sentences

Possible goals of a parser:

Find all possible trees, maybe ranked by probability or find the most likely tree.

Parsing performance depends on many aspects:

- Grammar expressivity (combination of symbols)
Parsing Natural Language Sentences

Possible goals of a parser:

Find all possible trees, maybe ranked by probability or find the most likely tree.

Parsing performance depends on many aspects:

- Grammar expressivity (combination of symbols)
- Coverage (words)
Parsing Natural Language Sentences

Possible goals of a parser:

- Find all possible trees, maybe ranked by probability or find the most likely tree.

Parsing performance depends on many aspects:

- Grammar expressivity (combination of symbols)
- Coverage (words)
- Parsing strategy (bottom-up, top-down)
Possible goals of a parser:

Find all possible trees, maybe ranked by probability or find the most likely tree.

Parsing performance depends on many aspects:

- Grammar expressivity (combination of symbols)
- Coverage (words)
- Parsing strategy (bottom-up, top-down)
- Rule application order (largest rule, most likely rule)
Possible goals of a parser:

Find all possible trees, maybe ranked by probability or find the most likely tree.

Parsing performance depends on many aspects:

- Grammar expressivity (combination of symbols)
- Coverage (words)
- Parsing strategy (bottom-up, top-down)
- Rule application order (largest rule, most likely rule)
- Ambiguity management (keep all, select one - probabilities, semantics, pragmatics)
- ...
The problem of repeating derivations

- Top-down and bottom-up strategies both lead to repeated derivations when using backtracking

Ex: ”a flight from Indianapolis to Houston [on TWA...]”

\[ \text{NG} \rightarrow \text{NN} \]
\[ \text{NG} \rightarrow \text{NG PP} \]
The problem of repeating derivations

- Top-down and bottom-up strategies both lead to repeated derivations when using backtracking

Ex: "a flight from Indianapolis to Houston"

NG → NN
NG → NG PP

```
NP
  / 
 DT  NG
    / 
   a  NG
      / 
     NG PP
        / 
       NG PP
          / 
         NG
           / 
          NN
           flight
        / 
       PP

   to Houston

from Indianapolis
```
Outline

1. Syntactic parsing
   - Goal and motivation

2. Trees and Grammars

3. Constituent Parsing
   - Background
   - Chart-based methods
   - CKY Algorithm
Properties

- They avoid re-doing derivations using dynamic programming.
- They represent derivations as a directed graph named chart.
- They use a dynamic programming table to build the chart.
Chart

- Nodes: positions between words of the input sentence
- Edges: **dotted rules** subsuming a sequence of words of the input sentence

- Dotted rules represent rules states:
  - Passive rules: $A \rightarrow B_1 \ldots B_k \bullet$
  - Active rules: $A \rightarrow B_1 \ldots B_i \bullet B_{i+1} \ldots B_k$

**Ex:**
Chart as a dynamic programming table

Syntactic parsing
Trees and Grammars
Constituent Parsing
Chart-based methods

Chart as a dynamic programming table for the sentence: "the cat eats fish"
Popular chart-based algorithms

- **CKY algorithm**
  - introduced dynamic programming
  - limited to CFGs in Chomsky Normal Form
  - passive bottom-up chart parser (only passive rules)
  - straightforward probabilistic version

- **Earley algorithm**
  - any CFG
  - active top-down parser (active/passive rules)
  - non-straightforward probabilistic version

- **Generalized chart parsing**
Outline

1. Syntactic parsing
   - Goal and motivation

2. Trees and Grammars

3. Constituent Parsing
   - Background
   - Chart-based methods
   - CKY Algorithm
CKY Algorithm properties

- Bottom-up
- Requires a grammar in Chomsky Normal Form (CNF).
- Dynamic programming: Store partial results that can be reused in different candidate solutions.
- Analogous to Viterbi in HMMs.
- Intermediate results stored in a chart structure.
A CFG $G = (N, \Sigma, R, S)$ expressed in CNF is as follows:

- $N$ is a set of non-terminal symbols
- $\Sigma$ is a set of terminal symbols
- $R$ is a set of rules which take one of two forms:
  - $X \rightarrow Y_1 Y_2$ for $X, Y_1, Y_2 \in N$
  - $X \rightarrow \alpha$ for $X \in N$ and $\alpha \in \Sigma$
- $S \in N$ is a start symbol

Any CFG can be converted into CNF
CKY Algorithm

Chart content:

- Maximum probability of a subtree with root X spanning words $i \ldots j$:
  \[ \pi(i, j, X) \]

- Backpath to recover which rules produced the maximum probability tree:
  \[ \psi(i, j, X) \]

The goal is to compute:

- $\max_{t \in \mathcal{T}(s)} p(t) = \pi(1, n, S)$
- $\psi(1, n, S)$

- It is possible to use it without probabilities to get all parse trees (with higher complexity)
CKY Algorithm

Base case: Tree leaves

- $\forall i = 1 \ldots n, \forall X \rightarrow w_i \in R, \pi(i, i, X) = q(X \rightarrow w_i)$

Recursive case: Non-terminal nodes

- $\forall i = 1 \ldots n, \forall j = (i + 1) \ldots n, \forall X \in N$
  $\pi(i, j, X) = \max_{X \rightarrow YZ \in R, k: i \leq k < j} q(X \rightarrow YZ) \times \pi(i, k, Y) \times \pi(k + 1, j, Z)$

$\psi(i, j, X) = \arg \max_{X \rightarrow YZ \in R, k: i \leq k < j} q(X \rightarrow YZ) \times \pi(i, k, Y) \times \pi(k + 1, j, Z)$

Output:
- Return $\pi(1, n, S)$ and recover backpath through $\psi(1, n, S)$
CKY Algorithm - Example

\[ N = \{S, \text{VP, NP, PP, DT, Vi, Vt, NN, IN}\} \]

\[ S = \{S\} \]

\[ \Sigma = \{\text{sleeps, saw, man, woman, telescope, the, with, in}\} \]

\[ R = \begin{cases} 
S \rightarrow \text{NP VP} & 1.0 \\
\text{NP} \rightarrow \text{DT NN} & 0.4 \\
\text{NP} \rightarrow \text{NP PP} & 0.6 \\
\text{PP} \rightarrow \text{IN NP} & 1.0 \\
\text{VP} \rightarrow \text{Vi} & 0.5 \\
\text{VP} \rightarrow \text{Vt NP} & 0.4 \\
\text{VP} \rightarrow \text{VP PP} & 0.1 \\
\end{cases} \]

1. \(S\)=sentence, \(\text{VP}\)=verb phrase, \(\text{NP}\)=noun phrase, \(\text{PP}\)=prepositional phrase, \(\text{DT}\)=determiner, \(\text{Vi}\)=intransitive verb, \(\text{Vt}\)=transitive verb, \(\text{NN}\)=noun, \(\text{IN}\)=preposition
### CKY Algorithm - Example - CNF

- **N** = \{S, VP, NP, PP, DT, Vi, Vt, NN, IN\}
- **S** = \{S\}
- **Σ** = \{sleeps, saw, man, woman, telescope, the, with, in\}

\[ R = \begin{cases} 
S \rightarrow \text{NP } \text{VP} & 1.0 \\
\text{NP} \rightarrow \text{DT } \text{NN} & 0.4 \\
\text{NP} \rightarrow \text{NP } \text{PP} & 0.6 \\
\text{PP} \rightarrow \text{IN } \text{NP} & 1.0 \\
\text{VP} \rightarrow \text{Vi} & 0.5 \\
\text{VP} \rightarrow \text{Vt } \text{NP} & 0.4 \\
\text{VP} \rightarrow \text{VP } \text{PP} & 0.1 \\
\end{cases} \]

1. S=sentence, VP=verb phrase, NP=noun phrase, PP=prepositional phrase, DT=determiner, Vi=intransitive verb, Vt=transitive verb, NN=noun, IN=preposition
CKY Algorithm - Example - CNF

\[ N = \{ S, \text{VP}, \text{NP}, \text{PP}, \text{DT}, \text{Vi}, \text{Vt}, \text{NN}, \text{IN} \}^1 \]

\[ S = \{ S \} \]

\[ \Sigma = \{ \text{sleeps, saw, man, woman, telescope, the, with, in} \} \]

\[ R = \begin{cases} 
S \rightarrow \text{NP} \ \text{VP} & 0.5 \\
S \rightarrow \text{NP} \ \text{Vi} & 0.5 \\
\text{NP} \rightarrow \text{DT} \ \text{NN} & 0.4 \\
\text{NP} \rightarrow \text{NP} \ \text{PP} & 0.6 \\
\text{PP} \rightarrow \text{IN} \ \text{NP} & 1.0 \\
\text{VP} \rightarrow \text{Vt} \ \text{NP} & 0.4 \\
\text{VP} \rightarrow \text{VP} \ \text{PP} & 0.1 \\
\text{VP} \rightarrow \text{Vi} \ \text{PP} & 0.5 \\
\end{cases} \]

\[ 1S=\text{sentence, VP=verb phrase, NP=noun phrase, PP=prepositional phrase, DT=determiner, Vi=intransitive verb, Vt=transitive verb, NN=noun, IN=preposition} \]
CKY Algorithm - Example

S \rightarrow NP \ VP \quad 0.5 \quad Vi \rightarrow \text{sleeps} \quad 1.0
S \rightarrow NP \ Vi \quad 0.5 \quad Vt \rightarrow \text{saw} \quad 1.0
NP \rightarrow DT \ NN \quad 0.4 \quad NN \rightarrow \text{man} \quad 0.7
NP \rightarrow NP \ PP \quad 0.6 \quad NN \rightarrow \text{woman} \quad 0.2
PP \rightarrow IN \ NP \quad 1.0 \quad NN \rightarrow \text{telescope} \quad 0.1
VP \rightarrow Vt \ NP \quad 0.4 \quad DT \rightarrow \text{the} \quad 1.0
VP \rightarrow VP \ PP \quad 0.1 \quad IN \rightarrow \text{with} \quad 0.5
VP \rightarrow Vi \ PP \quad 0.5 \quad IN \rightarrow \text{in} \quad 0.5
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw  1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in  0.5

Syntactic parsing

Trees and Grammars

Constituent Parsing

CKY Algorithm
CKY Algorithm - Example

- S $\rightarrow$ NP VP $0.5$
- S $\rightarrow$ NP Vi $0.5$
- NP $\rightarrow$ DT NN $0.4$
- NP $\rightarrow$ NP PP $0.6$
- PP $\rightarrow$ IN NP $1.0$
- VP $\rightarrow$ Vt NP $0.4$
- VP $\rightarrow$ VP PP $0.1$
- VP $\rightarrow$ Vi PP $0.5$

- Vi $\rightarrow$ sleeps $1.0$
- Vt $\rightarrow$ saw $1.0$
- NN $\rightarrow$ man $0.7$
- NN $\rightarrow$ woman $0.2$
- NN $\rightarrow$ telescope $0.1$
- DT $\rightarrow$ the $1.0$
- IN $\rightarrow$ with $0.5$
- IN $\rightarrow$ in $0.5$

**Examples of CKY Algorithm Calculations:**

1. $S: NP$ $\rightarrow$ DT NN $0.4 \times 1.0 \times 0.2 = 0.08$
2. $NP$ $\rightarrow$ DT $0.1 \times 1.0 \times 0.02 = 0.002$
3. $NN$ $\rightarrow$ man $0.4 \times 1.0 \times 0.7 = 0.28$
4. $NN$ $\rightarrow$ woman $0.4 \times 1.0 \times 0.2 = 0.08$
5. $NN$ $\rightarrow$ telescope $0.4 \times 1.0 \times 0.1 = 0.04$

**Example Calculation:**

- $S \rightarrow NP$ $0.5 \times 1 = 0.5$
- $S \rightarrow NP$ $0.5 \times 1 = 0.5$
- $NP \rightarrow DT$ $0.4 \times 1 = 0.4$
- $NP \rightarrow NP$ $0.6 \times 1 = 0.6$
- $PP \rightarrow IN$ $1 \times 1 = 1$
- $VP \rightarrow Vt$ $0.4 \times 1 = 0.4$
- $VP \rightarrow VP$ $0.1 \times 1 = 0.1$
- $VP \rightarrow Vi$ $0.5 \times 1 = 0.5$

**Example Parsing Path:**

- $S$ $\rightarrow$ NP VP $0.5$
- $NP$ $\rightarrow$ DT NN $0.4$
- $NN$ $\rightarrow$ man $0.7$
- $VP$ $\rightarrow$ Vt NP $0.4$
- $Vt$ $\rightarrow$ saw $1.0$
- $NP$ $\rightarrow$ NP PP $0.6$
- $IN$ $\rightarrow$ with $0.5$
- $PP$ $\rightarrow$ IN NP $1.0$
- $IN$ $\rightarrow$ in $0.5$
- $VP$ $\rightarrow$ VP PP $0.1$
- $DT$ $\rightarrow$ the $1.0$
- $NN$ $\rightarrow$ woman $0.2$
- $NN$ $\rightarrow$ telescope $0.1$

**Example Matrix:**

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
</tr>
<tr>
<td>2</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
</tr>
<tr>
<td>3</td>
<td>Vt</td>
<td>Vt</td>
<td>Vt</td>
<td>Vt</td>
<td>Vt</td>
<td>Vt</td>
<td>Vt</td>
<td>Vt</td>
</tr>
<tr>
<td>4</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
</tr>
<tr>
<td>5</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
</tr>
<tr>
<td>6</td>
<td>IN</td>
<td>IN</td>
<td>IN</td>
<td>IN</td>
<td>IN</td>
<td>IN</td>
<td>IN</td>
<td>IN</td>
</tr>
<tr>
<td>7</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
<td>DT</td>
</tr>
<tr>
<td>8</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
<td>NN</td>
</tr>
</tbody>
</table>
CKY Algorithm - Example

S → NP VP 0.5 Vi → sleeps 1.0
S → NP Vi 0.5 Vt → saw 1.0
NP → DT NN 0.4 NN → man 0.7
NP → NP PP 0.6 NN → woman 0.2
PP → IN NP 1.0 NN → telescope 0.1
VP → Vt NP 0.4 DT → the 1.0
VP → VP PP 0.1 IN → with 0.5
VP → Vi PP 0.5 IN → in 0.5
CKY Algorithm - Example

S → NP VP  0.5  Vi → sleeps  1.0
S → NP Vi  0.5  Vt → saw  1.0
NP → DT NN 0.4  NN → man  0.7
NP → NP PP 0.6  NN → woman  0.2
PP → IN NP 1.0  NN → telescope  0.1
VP → Vt NP 0.4  DT → the  1.0
VP → VP PP 0.1  IN → with  0.5
VP → Vi PP 0.5  IN → in  0.5
**CKY Algorithm - Example**

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
<th>Word</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S \rightarrow NP\ VP$</td>
<td>0.5</td>
<td>$Vi \rightarrow$ sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>$S \rightarrow NP\ Vi$</td>
<td>0.5</td>
<td>$Vt \rightarrow$ saw</td>
<td>1.0</td>
</tr>
<tr>
<td>$NP \rightarrow DT\ NN$</td>
<td>0.4</td>
<td>$NN \rightarrow$ man</td>
<td>0.7</td>
</tr>
<tr>
<td>$NP \rightarrow NP\ PP$</td>
<td>0.6</td>
<td>$NN \rightarrow$ woman</td>
<td>0.2</td>
</tr>
<tr>
<td>$PP \rightarrow IN\ NP$</td>
<td>1.0</td>
<td>$NN \rightarrow$ telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>$VP \rightarrow Vt\ NP$</td>
<td>0.4</td>
<td>$DT \rightarrow$ the</td>
<td>1.0</td>
</tr>
<tr>
<td>$VP \rightarrow VP\ PP$</td>
<td>0.1</td>
<td>$IN \rightarrow$ with</td>
<td>0.5</td>
</tr>
<tr>
<td>$VP \rightarrow Vi\ PP$</td>
<td>0.5</td>
<td>$IN \rightarrow$ in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**CKY Algorithm**

```
11  DT → the
    1.0
22  NN → woman
    0.2
33  Vt → saw
    1.0
44  DT → the
    1.0
55  NN → man
    0.7
66  IN → with
    0.5
77  DT → the
    1.0
88  NN → telescope
    0.1
```
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
CKY Algorithm - Example

S → NP VP     0.5      Vi → sleeps   1.0
S → NP Vi      0.5      Vt → saw      1.0
NP → DT NN     0.4      NN → man      0.7
NP → NP PP     0.6      NN → woman    0.2
PP → IN NP     1.0      NN → telescope 0.1
VP → Vt NP     0.4      DT → the      1.0
VP → VP PP     0.1      IN → with     0.5
VP → Vi PP     0.5      IN → in       0.5
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
### CKY Algorithm - Example

<table>
<thead>
<tr>
<th>Production</th>
<th>Probability</th>
<th>Token</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S \rightarrow NP \ VP$</td>
<td>0.5</td>
<td>sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>$S \rightarrow NP \ Vi$</td>
<td>0.5</td>
<td>saw</td>
<td>1.0</td>
</tr>
<tr>
<td>$NP \rightarrow DT \ NN$</td>
<td>0.4</td>
<td>man</td>
<td>0.7</td>
</tr>
<tr>
<td>$NP \rightarrow NP \ PP$</td>
<td>0.6</td>
<td>woman</td>
<td>0.2</td>
</tr>
<tr>
<td>$PP \rightarrow IN \ NP$</td>
<td>1.0</td>
<td>telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>$VP \rightarrow Vt \ NP$</td>
<td>0.4</td>
<td>the</td>
<td>1.0</td>
</tr>
<tr>
<td>$VP \rightarrow VP \ PP$</td>
<td>0.1</td>
<td>with</td>
<td>0.5</td>
</tr>
<tr>
<td>$VP \rightarrow Vi \ PP$</td>
<td>0.5</td>
<td>in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

#### CKY Algorithm Example

<table>
<thead>
<tr>
<th>Index</th>
<th>Token</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>3</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>4</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>5</td>
<td>NN</td>
<td>0.7</td>
</tr>
<tr>
<td>6</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>7</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>8</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>9</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>10</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>11</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>12</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>13</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>14</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>15</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>16</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>17</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>18</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>19</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>20</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>21</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>22</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>23</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>24</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>25</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>26</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>27</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>28</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>29</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>30</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>31</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>32</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>33</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>34</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>35</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>36</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>37</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>38</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>39</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>40</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>41</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>42</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>43</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>44</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>45</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>46</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>47</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>48</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>49</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>50</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>51</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>52</td>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>53</td>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>54</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>55</td>
<td>NN</td>
<td>0.7</td>
</tr>
<tr>
<td>56</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>57</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>58</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>59</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>60</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>61</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>62</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>63</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>64</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>65</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>66</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>67</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>68</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>69</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>70</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>71</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>72</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>73</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>74</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>75</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>76</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>77</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>78</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>79</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>80</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>81</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>82</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>83</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>84</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>85</td>
<td>NN</td>
<td>0.1</td>
</tr>
<tr>
<td>86</td>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>87</td>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>88</td>
<td>NN</td>
<td>0.1</td>
</tr>
</tbody>
</table>
### CKY Algorithm - Example

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S \rightarrow NP\ VP$</td>
<td>0.5</td>
</tr>
<tr>
<td>$S \rightarrow NP\ Vi$</td>
<td>0.5</td>
</tr>
<tr>
<td>$NP \rightarrow DT\ NN$</td>
<td>0.4</td>
</tr>
<tr>
<td>$NP \rightarrow NP\ PP$</td>
<td>0.6</td>
</tr>
<tr>
<td>$PP \rightarrow IN\ NP$</td>
<td>1.0</td>
</tr>
<tr>
<td>$VP \rightarrow Vt\ NP$</td>
<td>0.4</td>
</tr>
<tr>
<td>$VP \rightarrow VP\ PP$</td>
<td>0.1</td>
</tr>
<tr>
<td>$VP \rightarrow Vi\ PP$</td>
<td>0.5</td>
</tr>
<tr>
<td>$Vi \rightarrow$ sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>$Vt \rightarrow saw$</td>
<td>1.0</td>
</tr>
<tr>
<td>$NN \rightarrow$ man</td>
<td>0.7</td>
</tr>
<tr>
<td>$NN \rightarrow$ woman</td>
<td>0.2</td>
</tr>
<tr>
<td>$NN \rightarrow$ telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>$DT \rightarrow$ the</td>
<td>1.0</td>
</tr>
<tr>
<td>$IN \rightarrow$ with</td>
<td>0.5</td>
</tr>
<tr>
<td>$IN \rightarrow$ in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**CKY Algorithm - Example**

```
S → NP VP 0.5  Vi → sleeps 1.0  
S → NP Vi 0.5  Vt → saw 1.0  
NP → DT NN 0.4  NN → man 0.7  
NP → NP PP 0.6  NN → woman 0.2  
PP → IN NP 1.0  NN → telescope 0.1  
VP → Vt NP 0.4  DT → the 1.0  
VP → VP PP 0.1  IN → with 0.5  
VP → Vi PP 0.5  IN → in 0.5  
```
**CKY Algorithm - Example**

<table>
<thead>
<tr>
<th>Rule</th>
<th>Weight 1</th>
<th>Weight 2</th>
<th>Weight 3</th>
<th>Weight 4</th>
<th>Weight 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>S → NP VP</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S → NP Vi</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NP → DT NN</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NP → NP PP</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP → IN NP</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VP → Vt NP</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VP → VP PP</td>
<td>0.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VP → Vi PP</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vi → sleeps</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vt → saw</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN → man</td>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN → woman</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NN → telescope</td>
<td>0.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IN → with</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IN → in</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### CKY Algorithm - Example

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
<th>Example</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S \rightarrow NP \ VP$</td>
<td>0.5</td>
<td>Vi \rightarrow sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>$S \rightarrow NP \ Vi$</td>
<td>0.5</td>
<td>Vt \rightarrow saw</td>
<td>1.0</td>
</tr>
<tr>
<td>$NP \rightarrow DT \ NN$</td>
<td>0.4</td>
<td>NN \rightarrow man</td>
<td>0.7</td>
</tr>
<tr>
<td>$NP \rightarrow NP \ PP$</td>
<td>0.6</td>
<td>NN \rightarrow woman</td>
<td>0.2</td>
</tr>
<tr>
<td>$PP \rightarrow IN \ NP$</td>
<td>1.0</td>
<td>NN \rightarrow telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>$VP \rightarrow Vt \ NP$</td>
<td>0.4</td>
<td>DT \rightarrow the</td>
<td>1.0</td>
</tr>
<tr>
<td>$VP \rightarrow VP \ PP$</td>
<td>0.1</td>
<td>IN \rightarrow with</td>
<td>0.5</td>
</tr>
<tr>
<td>$VP \rightarrow Vi \ PP$</td>
<td>0.5</td>
<td>IN \rightarrow in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**CKY Algorithm - Example**

```
S → NP VP 0.5     Vi → sleeps 1.0
S → NP Vi 0.5     Vt → saw 1.0
NP → DT NN 0.4     NN → man 0.7
NP → NP PP 0.6     NN → woman 0.2
PP → IN NP 1.0     NN → telescope 0.1
VP → Vt NP 0.4     DT → the 1.0
VP → VP PP 0.1     IN → with 0.5
VP → Vi PP 0.5     IN → in 0.5
```
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5

Syntactic parsing

Trees and Grammars

Constituent Parsing

CKY Algorithm
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
## CKY Algorithm - Example

<table>
<thead>
<tr>
<th>Production Rule</th>
<th>Probability</th>
<th>Word</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>S → NP VP</td>
<td>0.5</td>
<td>Vi → sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>S → NP Vt</td>
<td>0.5</td>
<td>Vt → saw</td>
<td>1.0</td>
</tr>
<tr>
<td>NP → DT NN</td>
<td>0.4</td>
<td>NN → man</td>
<td>0.7</td>
</tr>
<tr>
<td>NP → NP PP</td>
<td>0.6</td>
<td>NN → woman</td>
<td>0.2</td>
</tr>
<tr>
<td>PP → IN NP</td>
<td>1.0</td>
<td>NN → telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>VP → Vt NP</td>
<td>0.4</td>
<td>DT → the</td>
<td>1.0</td>
</tr>
<tr>
<td>VP → VP PP</td>
<td>0.1</td>
<td>IN → with</td>
<td>0.5</td>
</tr>
<tr>
<td>VP → Vt PP</td>
<td>0.5</td>
<td>IN → in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**CKY Tree Example:**

```
S 0.5
  NP 0.4
    DT 1.0
    the
  VP 0.6
    Vt 1.0
    saw
  NP 0.2
    NN 0.7
    woman
  PP 0.1
    IN 0.5
    with
```

**Probability Calculations:**

1. **NP:**
   - NP → DT NN (0.4 * 1.0 * 0.2 = 0.08)
   - NP → NP PP (0.6 * 0.28 * 0.1 = 0.06)
   - NP → IN NP (1.0 * 0.7 * 0.1 = 0.07)

2. **VP:**
   - VP → Vt NP (0.4 * 1.0 * 0.28 = 0.112)
   - VP → VP PP (0.1 * 0.112 * 0.02 = 0.0023)
   - VP → Vt PP (0.5 * 0.28 * 0.02 = 0.007)

3. **IN:**
   - IN → with (0.5 * 0.112 * 0.02 = 0.00112)
   - IN → in (0.5 * 0.28 * 0.02 = 0.0014)

The CKY Algorithm uses these probabilities to construct a parse tree.
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0  
S → NP Vi 0.5  Vt → saw 1.0  
NP → DT NN 0.4  NN → man 0.7  
NP → NP PP 0.6  NN → woman 0.2  
PP → IN NP 1.0  NN → telescope 0.1  
VP → Vt NP 0.4  DT → the 1.0  
VP → VP PP 0.1  IN → with 0.5  
VP → Vi PP 0.5  IN → in 0.5  

Syntactic parsing
Trees and Grammars
Constituent Parsing
CKY Algorithm
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5

### CKY Algorithm - Example

1. **NP → DT NN 0.4**  NN → man 0.7
2. **NP → NP PP 0.6**  NN → woman 0.2
3. **PP → IN NP 1.0**  NN → telescope 0.1
4. **VP → Vt NP 0.4**  DT → the 1.0
5. **VP → VP PP 0.1**  IN → with 0.5
6. **VP → Vi PP 0.5**  IN → in 0.5

**Syntactic Parsing**

**Trees and Grammars**

**Constituent Parsing**

**CKY Algorithm**
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
CKY Algorithm - Example

S → NP VP  0.5  Vi → sleeps  1.0
S → NP Vi  0.5  Vt → saw    1.0
NP → DT NN  0.4  NN → man  0.7
NP → NP PP  0.6  NN → woman 0.2
PP → IN NP  1.0  NN → telescope 0.1
VP → Vt NP  0.4  DT → the  1.0
VP → VP PP  0.1  IN → with  0.5
VP → Vi PP  0.5  IN → in    0.5
### CKY Algorithm - Example

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
<th>Word</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S \rightarrow NP\ VP$</td>
<td>0.5</td>
<td>sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>$S \rightarrow NP\ Vi$</td>
<td>0.5</td>
<td>saw</td>
<td>1.0</td>
</tr>
<tr>
<td>$NP \rightarrow DT\ NN$</td>
<td>0.4</td>
<td>man</td>
<td>0.7</td>
</tr>
<tr>
<td>$NP \rightarrow NP\ PP$</td>
<td>0.6</td>
<td>woman</td>
<td>0.2</td>
</tr>
<tr>
<td>$PP \rightarrow IN\ NP$</td>
<td>1.0</td>
<td>telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>$VP \rightarrow Vt\ NP$</td>
<td>0.4</td>
<td>the</td>
<td>1.0</td>
</tr>
<tr>
<td>$VP \rightarrow VP\ PP$</td>
<td>0.1</td>
<td>with</td>
<td>0.5</td>
</tr>
<tr>
<td>$VP \rightarrow Vi\ PP$</td>
<td>0.5</td>
<td>in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

#### Syntactic parsing

#### Trees and Grammars

#### Constituent Parsing

#### CKY Algorithm
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
CKY Algorithm - Example

\[
S \rightarrow \text{NP VP} \quad 0.5 \quad \text{Vi} \rightarrow \text{sleeps} \quad 1.0
\]
\[
S \rightarrow \text{NP Vi} \quad 0.5 \quad \text{Vt} \rightarrow \text{saw} \quad 1.0
\]
\[
\text{NP} \rightarrow \text{DT NN} \quad 0.4 \quad \text{NN} \rightarrow \text{man} \quad 0.7
\]
\[
\text{NP} \rightarrow \text{NP PP} \quad 0.6 \quad \text{NN} \rightarrow \text{woman} \quad 0.2
\]
\[
\text{PP} \rightarrow \text{IN NP} \quad 1.0 \quad \text{NN} \rightarrow \text{telescope} \quad 0.1
\]
\[
\text{VP} \rightarrow \text{Vt NP} \quad 0.4 \quad \text{DT} \rightarrow \text{the} \quad 1.0
\]
\[
\text{VP} \rightarrow \text{VP PP} \quad 0.1 \quad \text{IN} \rightarrow \text{with} \quad 0.5
\]
\[
\text{VP} \rightarrow \text{Vi PP} \quad 0.5 \quad \text{IN} \rightarrow \text{in} \quad 0.5
\]
CKY Algorithm - Example

S → NP VP 0.5  Vi → sleeps 1.0
S → NP Vi 0.5  Vt → saw 1.0
NP → DT NN 0.4  NN → man 0.7
NP → NP PP 0.6  NN → woman 0.2
PP → IN NP 1.0  NN → telescope 0.1
VP → Vt NP 0.4  DT → the 1.0
VP → VP PP 0.1  IN → with 0.5
VP → Vi PP 0.5  IN → in 0.5
**CKY Algorithm - Example**

- **S → NP VP** 0.5  **Vi → sleeps** 1.0
- **S → NP Vi** 0.5  **Vt → saw** 1.0
- **NP → DT NN** 0.4  **NN → man** 0.7
- **NP → NP PP** 0.6  **NN → woman** 0.2
- **PP → IN NP** 1.0  **NN → telescope** 0.1
- **VP → Vt NP** 0.4  **DT → the** 1.0
- **VP → VP PP** 0.1  **IN → with** 0.5
- **VP → Vi PP** 0.5  **IN → in** 0.5

**Syntactic parsing**

**Trees and Grammars**

**Constituent Parsing**

**CKY Algorithm**
### CKY Algorithm - Example

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
<th>Word</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>S → NP VP</td>
<td>0.5</td>
<td>Vi → sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>S → NP Vi</td>
<td>0.5</td>
<td>Vt → saw</td>
<td>1.0</td>
</tr>
<tr>
<td>NP → DT NN</td>
<td>0.4</td>
<td>NN → man</td>
<td>0.7</td>
</tr>
<tr>
<td>NP → NP PP</td>
<td>0.6</td>
<td>NN → woman</td>
<td>0.2</td>
</tr>
<tr>
<td>PP → IN NP</td>
<td>1.0</td>
<td>NN → telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>VP → Vt NP</td>
<td>0.4</td>
<td>DT → the</td>
<td>1.0</td>
</tr>
<tr>
<td>VP → VP PP</td>
<td>0.1</td>
<td>IN → with</td>
<td>0.5</td>
</tr>
<tr>
<td>VP → Vi PP</td>
<td>0.5</td>
<td>IN → in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
<th>Word</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>VP → Vt NP</td>
<td>0.4</td>
<td>DT → the</td>
<td>1.0</td>
</tr>
<tr>
<td>VP → VP PP</td>
<td>0.1</td>
<td>IN → with</td>
<td>0.5</td>
</tr>
<tr>
<td>VP → Vi PP</td>
<td>0.5</td>
<td>IN → in</td>
<td>0.5</td>
</tr>
</tbody>
</table>
**CKY Algorithm - Example**

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
<th>Phrase</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>S → NP VP</td>
<td>0.5</td>
<td>Vi → sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>S → NP Vi</td>
<td>0.5</td>
<td>Vt → saw</td>
<td>1.0</td>
</tr>
<tr>
<td>NP → DT NN</td>
<td>0.4</td>
<td>NN → man</td>
<td>0.7</td>
</tr>
<tr>
<td>NP → NP PP</td>
<td>0.6</td>
<td>NN → woman</td>
<td>0.2</td>
</tr>
<tr>
<td>PP → IN NP</td>
<td>1.0</td>
<td>NN → telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>VP → Vt NP</td>
<td>0.4</td>
<td>DT → the</td>
<td>1.0</td>
</tr>
<tr>
<td>VP → VP PP</td>
<td>0.1</td>
<td>IN → with</td>
<td>0.5</td>
</tr>
<tr>
<td>VP → Vi PP</td>
<td>0.5</td>
<td>IN → in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**Example Calculation**

\[
\text{NP} \rightarrow \text{DT} \text{NN}
\]

\[
\text{NP} \rightarrow \text{NP} \text{PP}
\]

\[
\text{VP} \rightarrow \text{VP} \text{PP}
\]

\[
\text{VP} \rightarrow \text{NP} \text{Vi}
\]

\[
\text{VP} \rightarrow \text{Vt} \text{NP}
\]

\[
\text{NP} \rightarrow \text{DT} \text{NN}
\]

\[
\text{NP} \rightarrow \text{NP} \text{PP}
\]

\[
\text{PP} \rightarrow \text{IN} \text{NP}
\]

\[
\text{VP} \rightarrow \text{Vt} \text{NP}
\]

\[
\text{VP} \rightarrow \text{Vt} \text{PP}
\]

\[
\text{VP} \rightarrow \text{Vi} \text{PP}
\]

**Example CKY Algorithm Table**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>NN</td>
<td>0.2</td>
</tr>
<tr>
<td>Vt</td>
<td>1.0</td>
</tr>
<tr>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>NN</td>
<td>0.7</td>
</tr>
<tr>
<td>IN</td>
<td>0.5</td>
</tr>
<tr>
<td>DT</td>
<td>1.0</td>
</tr>
<tr>
<td>NN</td>
<td>0.1</td>
</tr>
</tbody>
</table>
CKY Algorithm - Example

\[
\begin{align*}
S & \rightarrow \text{NP} \ \text{VP} \quad 0.5 \\
S & \rightarrow \text{NP} \ \text{Vi} \quad 0.5 \\
\text{NP} & \rightarrow \text{DT} \ \text{NN} \quad 0.4 \\
\text{NP} & \rightarrow \text{NP} \ \text{PP} \quad 0.6 \\
\text{PP} & \rightarrow \text{IN} \ \text{NP} \quad 1.0 \\
\text{VP} & \rightarrow \text{Vt} \ \text{NP} \quad 0.4 \\
\text{VP} & \rightarrow \text{VP} \ \text{PP} \quad 0.1 \\
\text{VP} & \rightarrow \text{Vi} \ \text{PP} \quad 0.5 \\
\text{Vi} & \rightarrow \text{sleeps} \quad 1.0 \\
\text{Vt} & \rightarrow \text{saw} \quad 1.0 \\
\text{NN} & \rightarrow \text{man} \quad 0.7 \\
\text{NN} & \rightarrow \text{woman} \quad 0.2 \\
\text{NN} & \rightarrow \text{telescope} \quad 0.1 \\
\text{DT} & \rightarrow \text{the} \quad 1.0 \\
\text{IN} & \rightarrow \text{with} \quad 0.5 \\
\text{IN} & \rightarrow \text{in} \quad 0.5
\end{align*}
\]
### CKY Algorithm - Example

<table>
<thead>
<tr>
<th>Rule</th>
<th>Probability</th>
<th>Term</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>S → NP VP</td>
<td>0.5</td>
<td>Vi → sleeps</td>
<td>1.0</td>
</tr>
<tr>
<td>S → NP Vi</td>
<td>0.5</td>
<td>Vt → saw</td>
<td>1.0</td>
</tr>
<tr>
<td>NP → DT NN</td>
<td>0.4</td>
<td>NN → man</td>
<td>0.7</td>
</tr>
<tr>
<td>NP → NP PP</td>
<td>0.6</td>
<td>NN → woman</td>
<td>0.2</td>
</tr>
<tr>
<td>PP → IN NP</td>
<td>1.0</td>
<td>NN → telescope</td>
<td>0.1</td>
</tr>
<tr>
<td>VP → Vt NP</td>
<td>0.4</td>
<td>DT → the</td>
<td>1.0</td>
</tr>
<tr>
<td>VP → VP PP</td>
<td>0.1</td>
<td>IN → with</td>
<td>0.5</td>
</tr>
<tr>
<td>VP → Vi PP</td>
<td>0.5</td>
<td>IN → in</td>
<td>0.5</td>
</tr>
</tbody>
</table>

#### Final result

The CKY Algorithm is used to parse the sentence: "The woman saw the man with the telescope.

- **Final result**
  - S → NP VP = 0.5 * 0.08 * 1.34e-3 = 5.38e-5
  - VP → Vt33 NP45 = 0.4 * 1.0 * 3.36e-3 = 1.34e-3
  - NP → NP45 PP = 0.6 * 0.28 * 0.02 = 3.36e-3
  - VP → Vi PP = 0.5 * 0.08 * 0.112 = 4.48e-3
  - NP → DT11 NN22 = 0.4 * 1.0 * 0.2 = 0.08
  - NP → DT44 NN55 = 0.4 * 1.0 * 0.7 = 0.28
  - DT → the = 1.0
  - NN → woman = 0.2
  - Vt → saw = 1.0
  - DT → the = 1.0
  - NN → man = 0.7
  - IN → with = 0.5
  - DT → the = 1.0
  - NN → telescope = 0.1
Context-free means *independent of the context*, i.e., assumes that any expansion of a non-terminal is applicable, regardless of the context in which it occurs.
Natural Language is not Context-Free

- NP expansion (for instance) is highly dependent on the parent of the NP

- Complete context independence is a too strong independence assumption for natural language.
Natural Language is not Context-Free

- The application of a rule may affect the applicability of others

```
NP  | VP  
PRP | PP  
She | flew from Indianapolis
```
```
NP  | VP  
PRP | PP  
She | flew from Indianapolis
```
Natural Language is not Context-Free

- May contain non-projective structures:

  *John saw the dog yesterday which was a Yorkshire Terrier*