# CS 481

## *Artificial Intelligence Language Understanding*

**February 16, 2023**

# Announcements / Reminders

- Please follow the Week 06 To Do List instructions
- PA #01 due on ~~Monday (02/20/23) at 11:59 PM CST~~ Thursday (02/23/23) at 11:59 PM CST

- Exam dates:
  - Midterm: 03/02/2023 during Thursday lecture time
  - Final: 04/27/2023 during Thursday lecture time

# Plan for Today

- Treebank
- Constituency parsing

# Treebank

- In linguistics, a treebank is a parsed text corpus that annotates syntactic or semantic sentence structure.

- Treebanks are often created on top of a corpus that has already been annotated with part-of-speech tags. In turn, treebanks are sometimes enhanced with semantic or other linguistic information.

# Penn Treebank

**https://web.archive.org/web/20131109202842/http://www.cis.upenn.edu/~treebank/**

**Available via NLTK**

# Context Free Grammar

**A context-free grammar $G$ is defined by:**

- **$N$: a set of non-terminal symbols (variables)**
- **$\Sigma$: a set of terminal symbols (disjoint from $N$)**
- **$R$: a set of rules or productions of the form $A \rightarrow \beta$, where:**
  - **$A$: a non-terminal symbol**
  - **$\beta$: a string of symbols from the infinite set of strings $(\Sigma \cup N)^*$**
- **$S$: a designated start symbol**

# Derivation: Formal Definition

**Derivation is a generalization of direct derivation:**

**Let** $\alpha_1, \alpha_2, \ldots, \alpha_m$ **are strings in the set** $(\Sigma \cup N)^*$**,**
**with** $m \geq 1$ **such that**

$$\alpha_1 \Rightarrow \alpha_2, \alpha_2 \Rightarrow \alpha_3, \ldots, \alpha_{m-1} \Rightarrow \alpha_m$$

**and we can say that** $\alpha_1$ **derives** $\alpha_\mu$ **, or** $\alpha_1 \overset{*}{\Rightarrow} \alpha_\mu$

# Language vs. Grammar

**Language $L_G$ generated by a grammar $G$ is the set of all strings composed of terminal symbols that can be derived from the designated start symbol $S$.**

$$L_G = \{ w \mid w \ \mathit{is\ in} \quad * \ \ \mathit{and} \ S \ \overset{*}{\phantom{\Rightarrow}} \ w \}$$

strings made up of terminals $\boxed{\text{derived}}$ from symbol $S$

# Parsing

**The task of determining the parts of speech, phrases, clauses, and their relationship to one another is called parsing.**

# The Concept of Constituency

**Groups of words that may behave as a single unit or phrase are called a constituent.**

# Constituents: Examples

**There exist different kinds of constituents:**

- **noun phrases (NP): the man, a boy with a hat, Illinois**
- **prepositional phrases (PP): with glasses, in the room**
- **verb phrases (VP): eat pasta, sleep, sleep soundly**
- **etc.**

# Constituents: Heads and Dependents

**Every phrase has a head:**

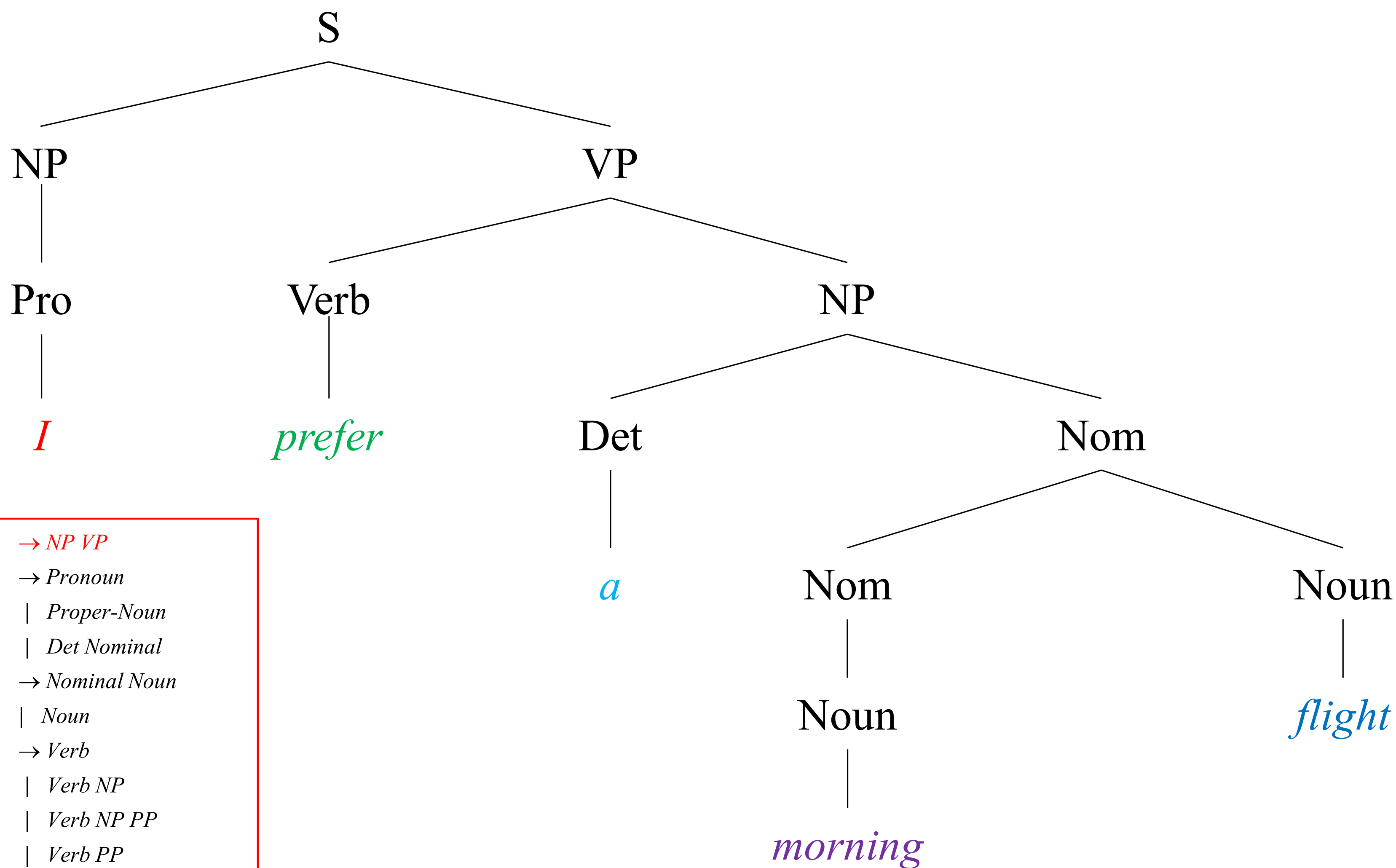
- **noun phrases (NP): the man, a boy with a hat, Illinois**
- **prepositional phrases (PP): with glasses, in the room**
- **verb phrases (VP): eat pasta, sleep, sleep soundly**

**all other parts are dependents (arguments or adjuncts)**

# Parse Tree: Example

**Parse tree (representing a sequence of expansions = derivation) for sentence:**

*I prefer a morning flight*




| | |
|---|---|
| *S* | → *NP VP* |
| *NP* | → *Pronoun* |
| | \| *Proper-Noun* |
| | \| *Det Nominal* |
| *Nominal* | → *Nominal Noun* |
| | \| *Noun* |
| *VP* | → *Verb* |
| | \| *Verb NP* |
| | \| *Verb NP PP* |
| | \| *Verb PP* |
| *NP* | → *Preposition NP* |

# Parse Tree: What Does It Tell Us?

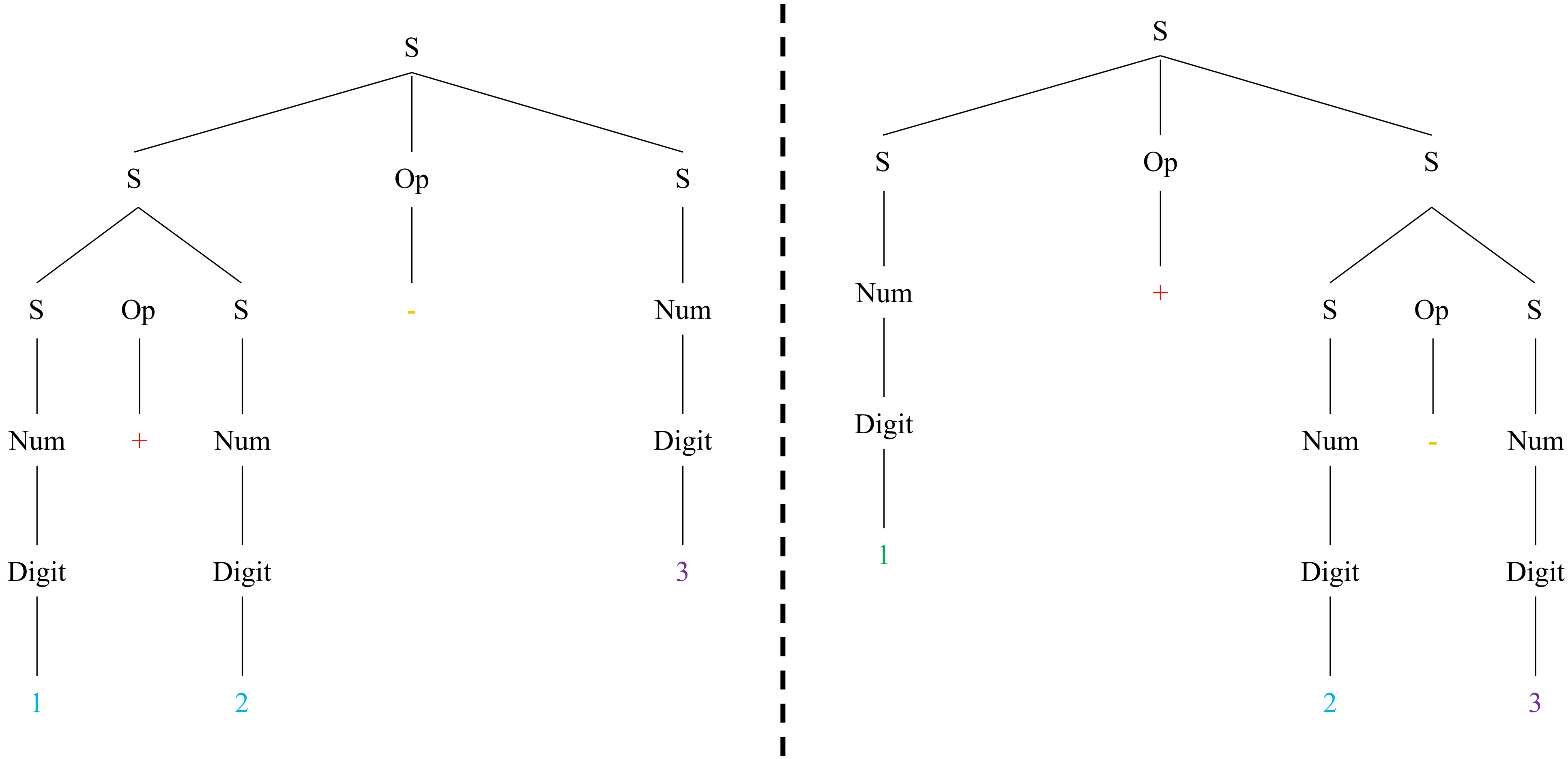
**If a parse tree for some sentence $s$ CAN be generated using grammar rules - what does it tell us about $s$?**

# Parse Tree: Applications

**Parse trees can be used:**

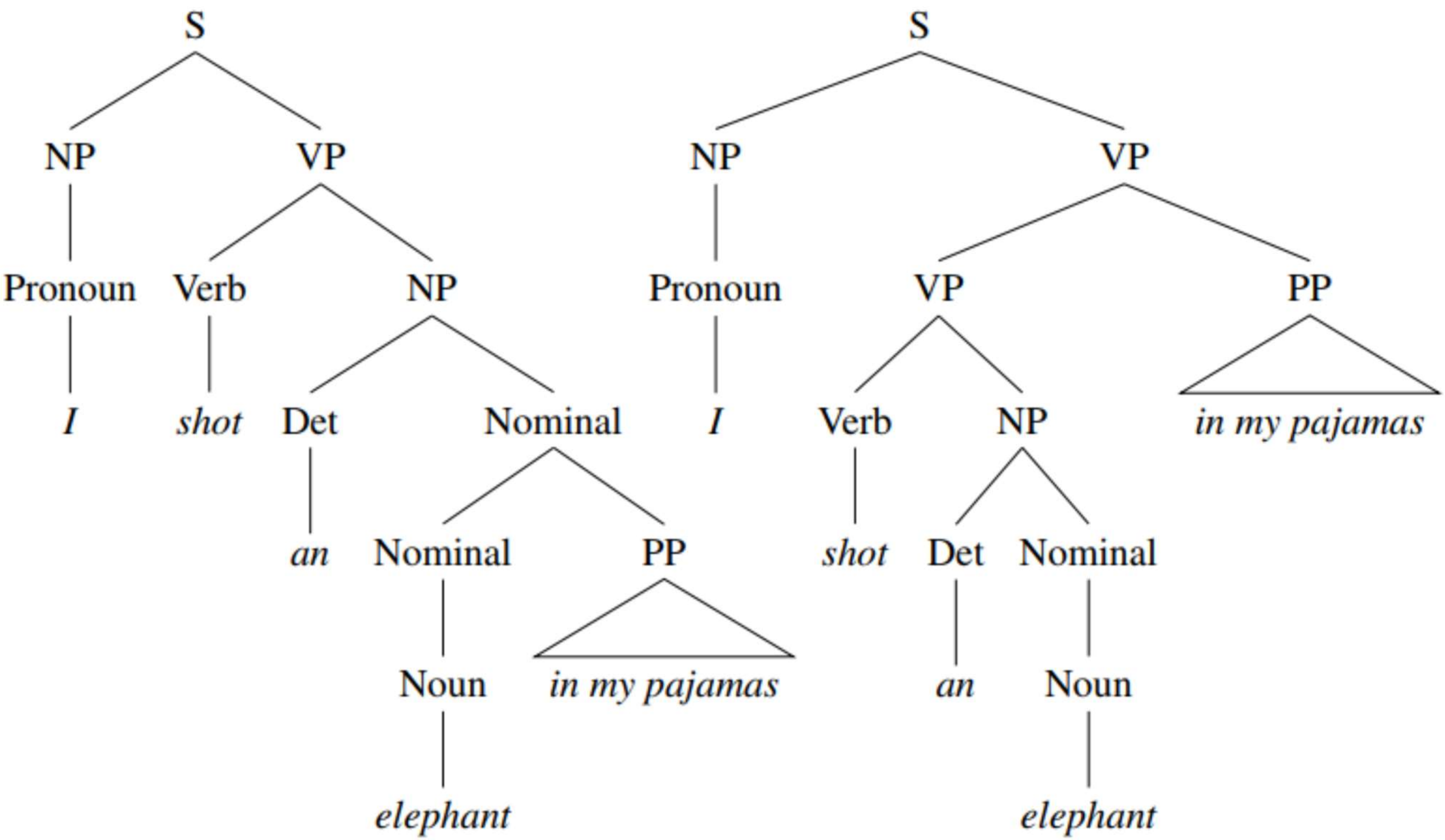
- **in grammar checking: sentence that cannot be parsed may have grammatical errors**
- **as an intermediate stage of representation for semantic analysis:**
  - **question answering**
  - **etc.**

# Ambiguous Grammar



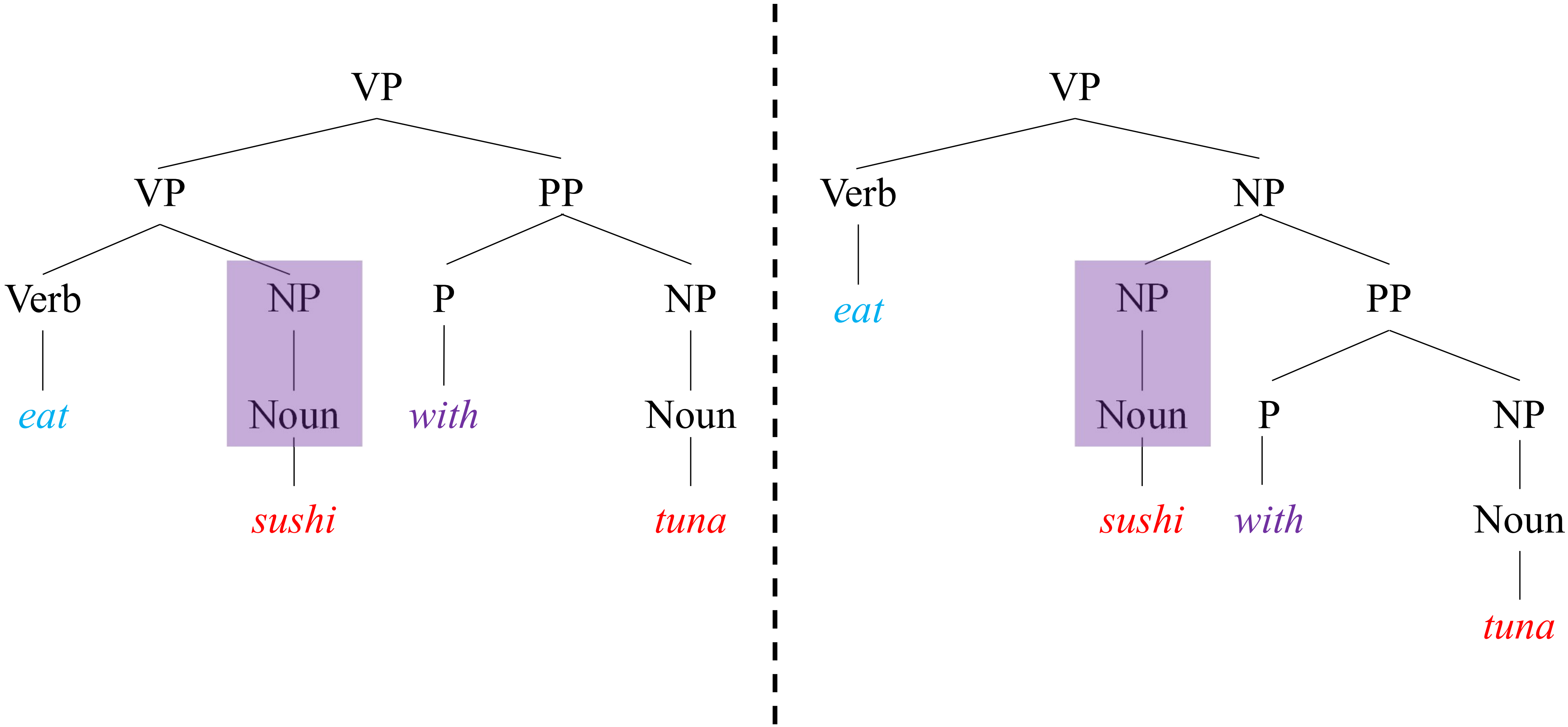

**A grammar is said to be ambiguous if it can generate the same string (here: 1 + 2 - 3) through multiple derivations.**

# Structural Ambiguity




**Structural ambiguity occurs when the grammar can assign more than one parse to a sentence.**

# Attachment Ambiguity




**A sentence has an attachment ambiguity if a particular constituent can be attached to the parse tree at more than one place.**

# Coordination Ambiguity

**Consider the expression:**

*Old men and women*

**It could be read as:**

*[Old [men and women]]*

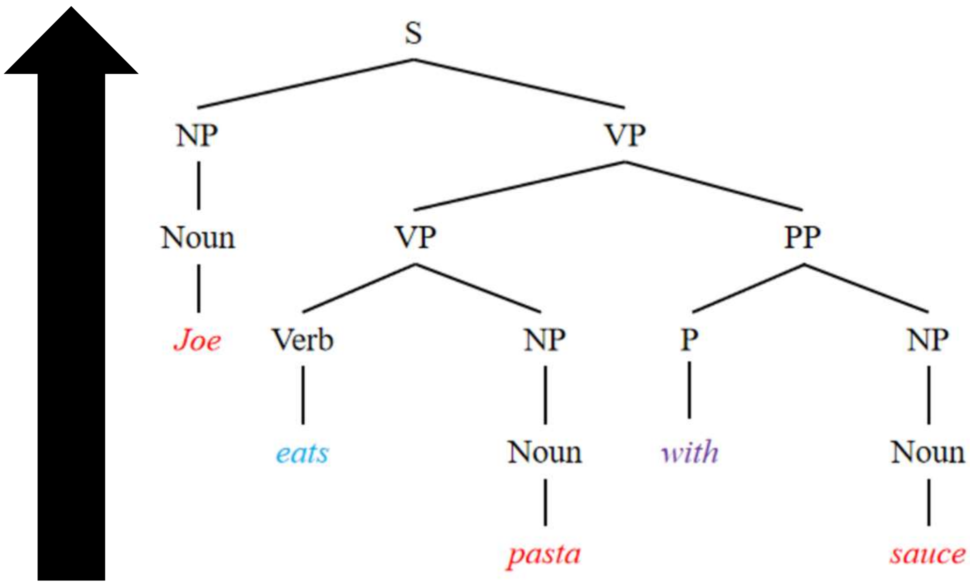
**It could be read as:**

*[Old men] and [women]*

**In coordination ambiguity phrases can be conjoined by a conjunction like *and*.**

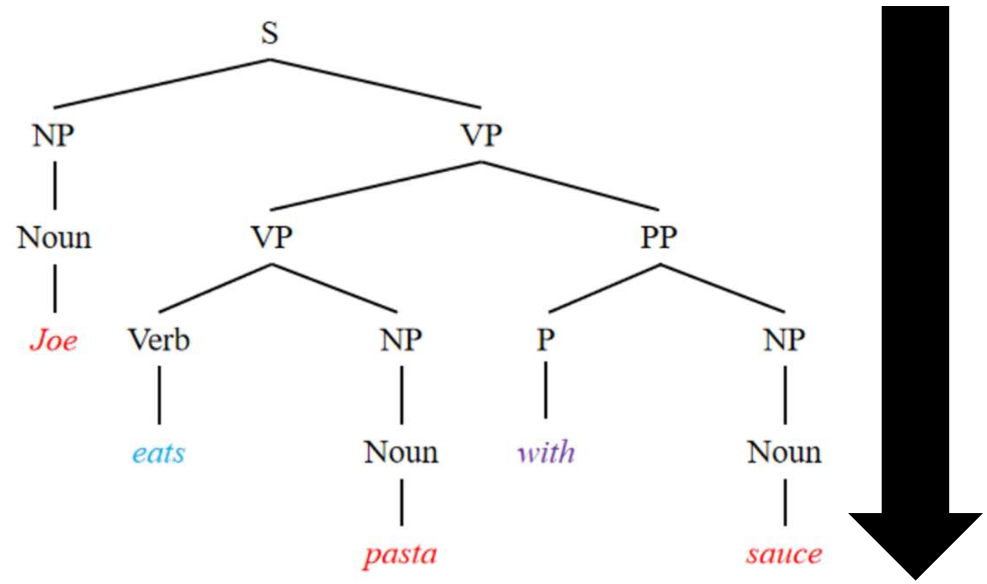
# Bottom-Up vs. Top-Down Parsing

## Bottom-Up Parsing




**Bottom-up parsing is a parsing technique that first looks at the lowest level of the parse tree and works up the parse tree by using the rules of grammar.**

## Top-Down Parsing




**Top-down parsing is a parsing technique that first looks at the highest level of the parse tree and works down the parse tree by using the rules of grammar.**

# Chomsky Normal Form

**A context-free grammar, $G$, is said to be in Chomsky normal form if all of its production rules are of the form:**

$$A \rightarrow BC \quad \textbf{or}$$

$$A \rightarrow a \quad \textbf{or}$$

$$S \rightarrow \varepsilon$$

**where:**

- $A$**,** $B$**, and** $C$ **are non-terminal symbols,**
- $a$ **is a terminal symbol**
- $S$ **is the start symbol, and** $\varepsilon$ **denotes the empty string.**
- **neither** $B$ **nor** $C$ **may be the start symbol**

# Conversion to Chomsky Normal Form

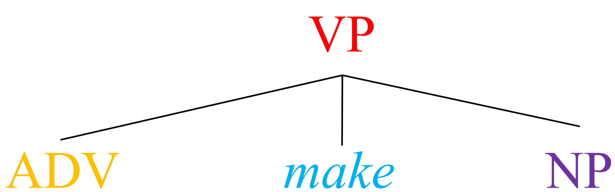
**The process of conversion of a standard contex-free grammar to its Chomsky Normal Form (CNF) can be summarized with:**

- **copy all conforming rules to the new grammar unchanged**
- **convert terminals within rules to dummy non-terminals**
- **convert unit productions**
- **make all rules binary and add them to new grammar**

# Conversion to Chomsky Normal Form

## Standard Grammar

$$\text{VP} \rightarrow \text{ADV}\ \textit{make}\ \text{NP}$$

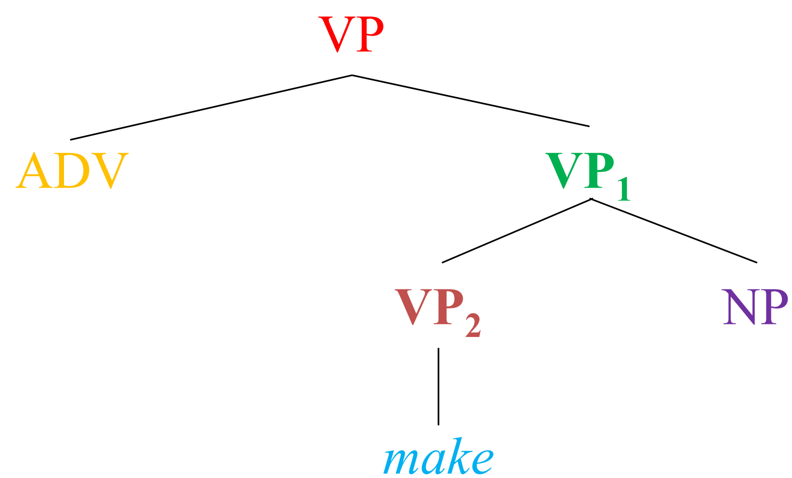



## Equivalent CNF Grammar

$$\text{VP} \rightarrow \text{ADV}\ \mathbf{VP_1}$$

$$\mathbf{VP_1} \rightarrow \mathbf{VP_2}\ \text{NP}$$

$$\mathbf{VP_2} \rightarrow \textit{make}$$



# Chomsky Normal Form Grammar

| $\mathscr{L}_1$ **Grammar** | $\mathscr{L}_1$ **in CNF** |
|---|---|
| $S \rightarrow NP\ VP$ | $S \rightarrow NP\ VP$ |
| $S \rightarrow Aux\ NP\ VP$ | $S \rightarrow X1\ VP$ |
| | $X1 \rightarrow Aux\ NP$ |
| $S \rightarrow VP$ | $S \rightarrow book \mid include \mid prefer$ |
| | $S \rightarrow Verb\ NP$ |
| | $S \rightarrow X2\ PP$ |
| | $S \rightarrow Verb\ PP$ |
| | $S \rightarrow VP\ PP$ |
| $NP \rightarrow Pronoun$ | $NP \rightarrow I \mid she \mid me$ |
| $NP \rightarrow Proper\text{-}Noun$ | $NP \rightarrow TWA \mid Houston$ |
| $NP \rightarrow Det\ Nominal$ | $NP \rightarrow Det\ Nominal$ |
| $Nominal \rightarrow Noun$ | $Nominal \rightarrow book \mid flight \mid meal \mid money$ |
| $Nominal \rightarrow Nominal\ Noun$ | $Nominal \rightarrow Nominal\ Noun$ |
| $Nominal \rightarrow Nominal\ PP$ | $Nominal \rightarrow Nominal\ PP$ |
| $VP \rightarrow Verb$ | $VP \rightarrow book \mid include \mid prefer$ |
| $VP \rightarrow Verb\ NP$ | $VP \rightarrow Verb\ NP$ |
| $VP \rightarrow Verb\ NP\ PP$ | $VP \rightarrow X2\ PP$ |
| | $X2 \rightarrow Verb\ NP$ |
| $VP \rightarrow Verb\ PP$ | $VP \rightarrow Verb\ PP$ |
| $VP \rightarrow VP\ PP$ | $VP \rightarrow VP\ PP$ |
| $PP \rightarrow Preposition\ NP$ | $PP \rightarrow Preposition\ NP$ |

# Chomsky Normal Form: Motivation

- **CNF grammar produces the same language as generated by its equivalent standard context-free grammar**
- **CNF grammar is used as a preprocessing step for many context-free grammar algorithms for CFG like CYK/CKY, bottom-up parsers, etc.**
- **generating string w of length ‘n’ requires ‘2n-1’ production or steps in CNF**

# CKY Parsing Algorithm

- **CKY: Cocke–Younger–Kasami**
- **Bottom-up parsing:**
  - **start with the words**
- **Dynamic programming:**
  - **save the results in a table/chart**
  - **re-use these results in finding larger constituents**
- **Complexity:** $O(\ n^3 * |G|\ )$
  - $n$: **length of string,** $|G|$: **size of grammar)**
- **Presumes a grammar in Chomsky Normal Form**

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

*Joe|**N** eats|**V** pasta|**N** with|**P** sauce|**N***

| Rule \| Production |
|---|
| S $\rightarrow$ NP VP |
| NP $\rightarrow$ Noun |
| NP $\rightarrow$ Det Noun |
| NP $\rightarrow$ NP PP |
| VP $\rightarrow$ Verb |
| VP $\rightarrow$ Verb NP |
| VP $\rightarrow$ VP PP |
| PP $\rightarrow$ P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | | | | | | *Joe* | 0 |
| | | | | | | *eats* | 1 |
| | | | | | | *pasta* | 2 |
| | | | | | | *with* | 3 |
| | | | | | | *sauce* | 4 |

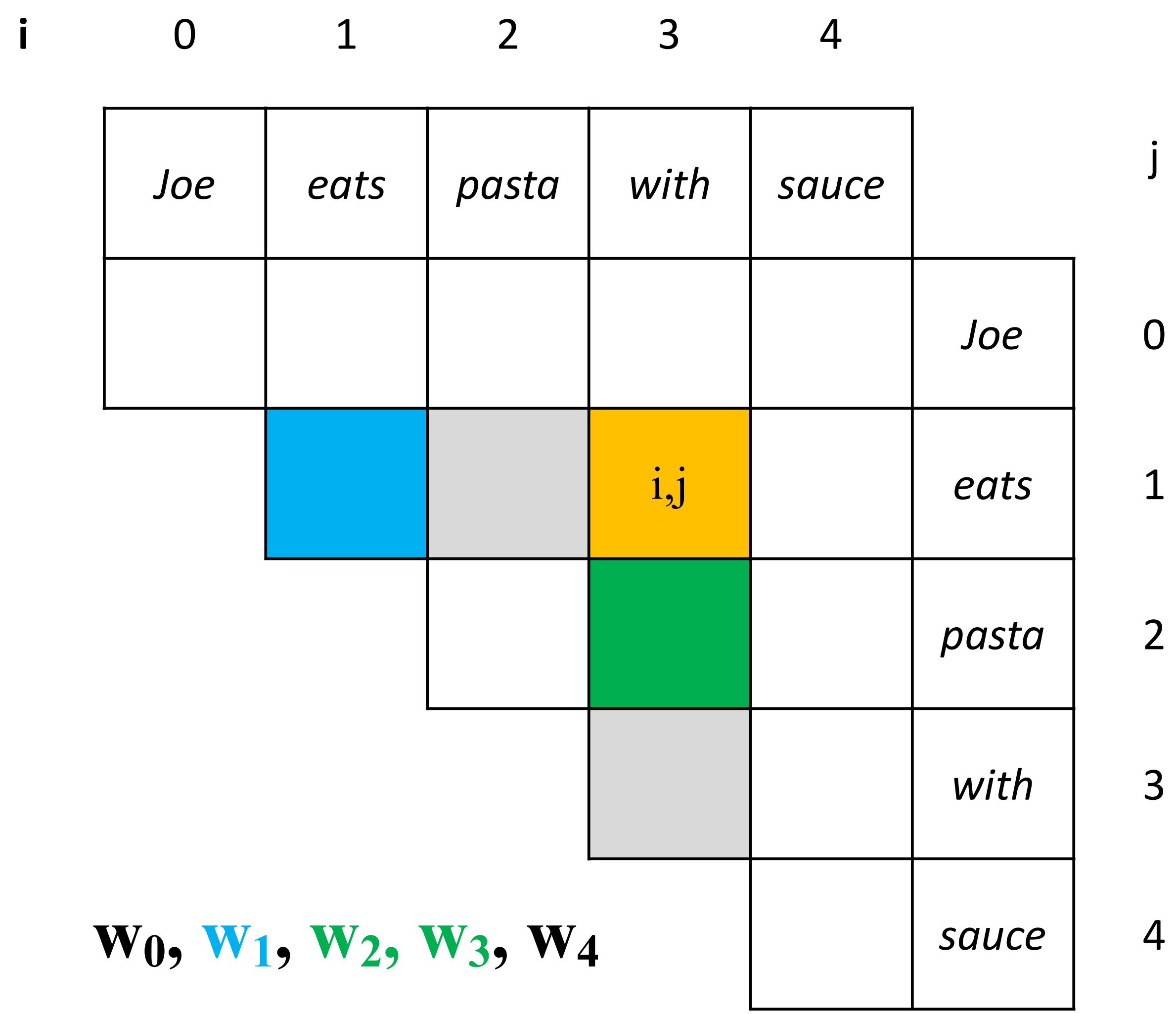
| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition: Sentence Splits

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***




**$w_0$, $w_1$, $w_2$, $w_3$, $w_4$**

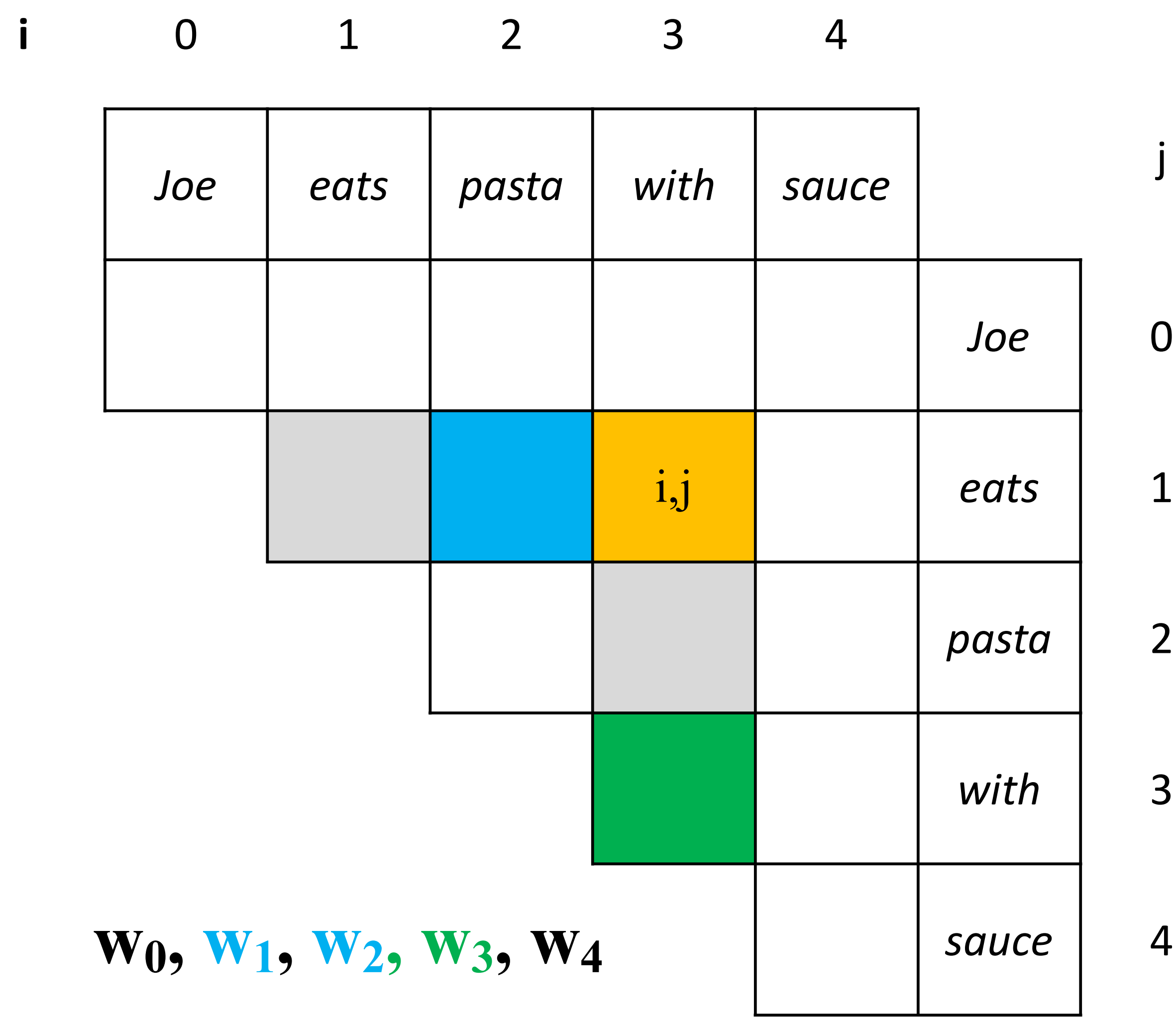
| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition: Sentence Splits

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe|**N*** $_1$ *eats|**V*** $_2$ *pasta|**N*** $_3$ *with|**P*** $_4$ *sauce|**N***



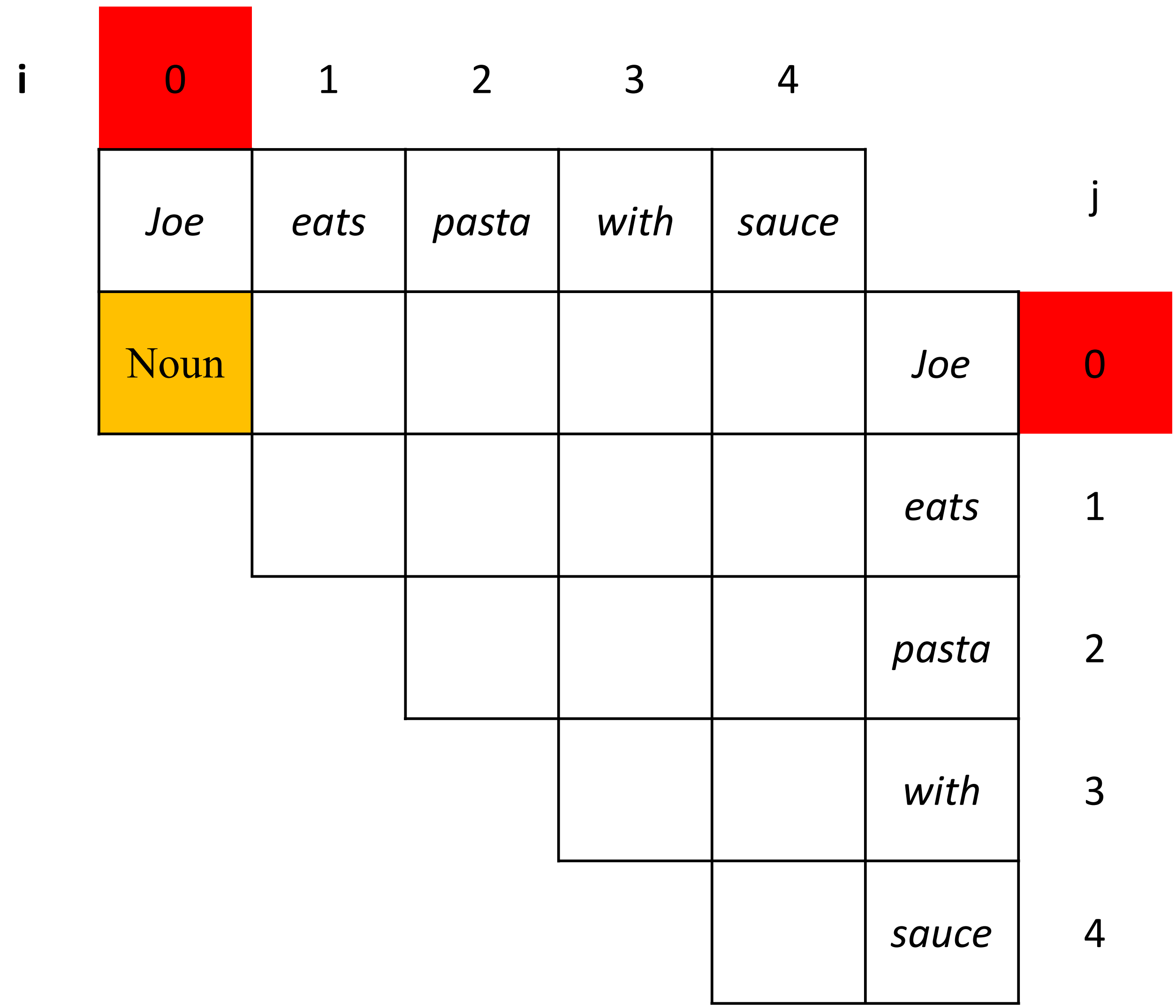

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***



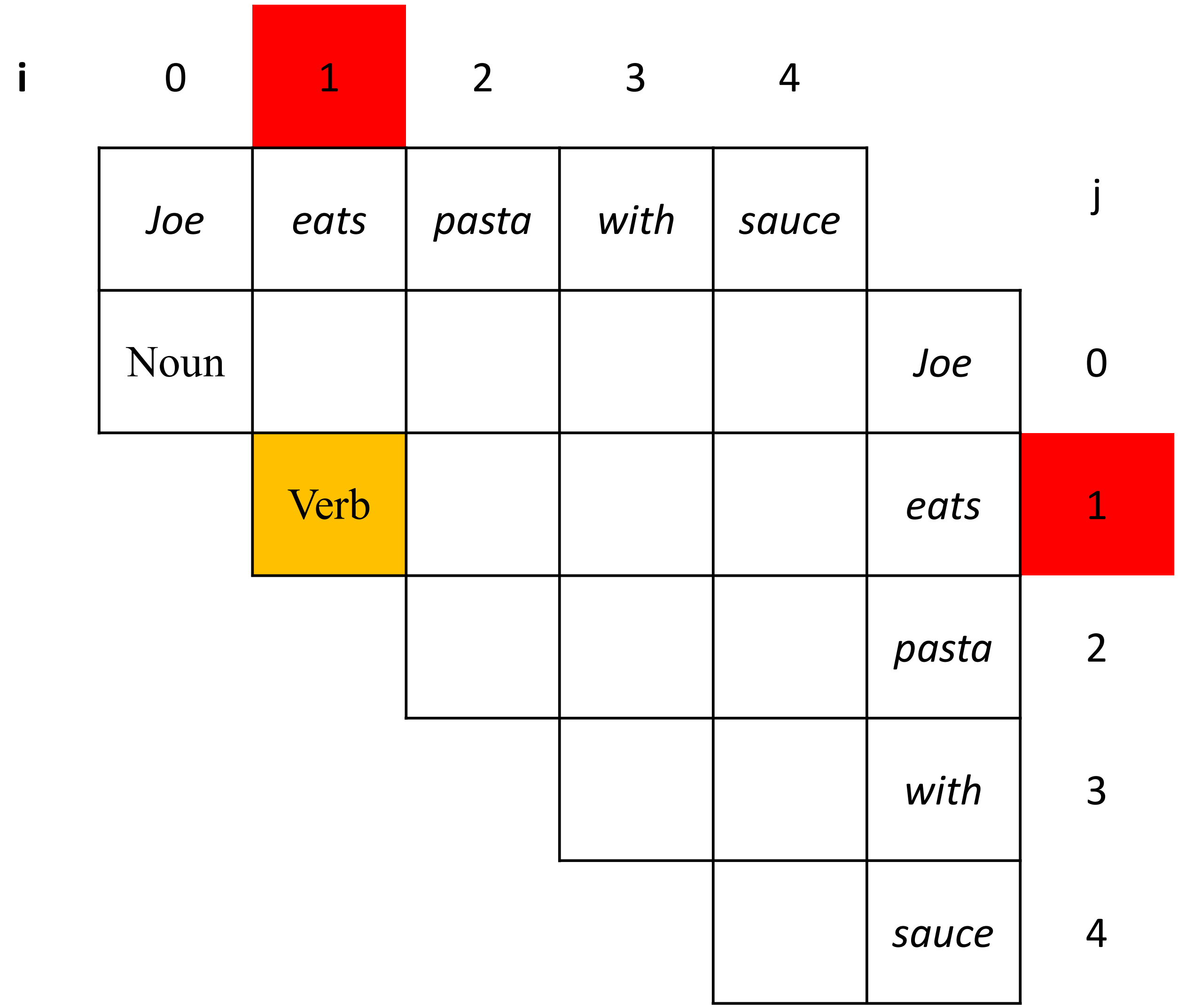

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_{0}$ *Joe*|***N*** $_{1}$ *eats*|***V*** $_{2}$ *pasta*|***N*** $_{3}$ *with*|***P*** $_{4}$ *sauce*|***N***




| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|**N** $_1$ *eats*|**V** $_2$ *pasta*|**N** $_3$ *with*|**P** $_4$ *sauce*|**N**

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | | | | | *Joe* | 0 |
| | | Verb | | | | *eats* | 1 |
| | | | Noun | | | *pasta* | 2 |
| | | | | | | *with* | 3 |
| | | | | | | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | | | | | *Joe* | 0 |
| | | Verb | | | | *eats* | 1 |
| | | | Noun | | | *pasta* | 2 |
| | | | | P | | *with* | 3 |
| | | | | | | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun | | | | | *Joe* | 0 |
| | | Verb | | | | *eats* | 1 |
| | | | Noun | | | *pasta* | 2 |
| | | | | P | | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | | | | *Joe* | 0 |
| | | Verb | | | | *eats* | 1 |
| | | | Noun | | | *pasta* | 2 |
| | | | | P | | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| **Rule \| Production** |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | | | | *Joe* | 0 |
| | | Verb | VP | | | *eats* | 1 |
| | | | Noun | | | *pasta* | 2 |
| | | | | P | | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | | | | *Joe* | 0 |
| | | Verb | VP | | | *eats* | 1 |
| | | | Noun | ∅ | | *pasta* | 2 |
| | | | | P | | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun | S | | | | *Joe* | 0 |
| | | Verb | VP | | | *eats* | 1 |
| | | | Noun | ∅ | | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | | | *Joe* | 0 |
| | | Verb | VP | | | *eats* | 1 |
| | | | Noun | ∅ | | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | ∅ | | | *Joe* | 0 |
| | | Verb | VP | | | *eats* | 1 |
| | | | Noun | ∅ | | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | | | *Joe* | 0 |
| | | Verb | VP | ∅ | | *eats* | 1 |
| | | | Noun | ∅ | | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | | | *Joe* | 0 |
| | | Verb | VP | ∅ | | *eats* | 1 |
| | | | Noun | ∅ | | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | | | *Joe* | 0 |
| | | Verb | VP | ∅ | | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| **Rule \| Production** |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | | | *Joe* | 0 |
| | | Verb | VP | ∅ | | *eats* | 1 |
| | | | Noun | ∅ | ∅ | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| **Rule \| Production** |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | | *Joe* | 0 |
| | | Verb | VP | ∅ | | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| **Rule \| Production** |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | | *Joe* | 0 |
| | | Verb | VP | ∅ | | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | | *Joe* | 0 |
| | | Verb | VP | ∅ | | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| **Rule \| Production** |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | | *Joe* | 0 |
| | | Verb | VP | ∅ | ∅ | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun | S | S | ∅ | S | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | ∅ | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|**N** $_1$ *eats*|**V** $_2$ *pasta*|**N** $_3$ *with*|**P** $_4$ *sauce*|**N**

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun | S | S | ∅ | ∅ | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | j |
| | Noun | S | S | ∅ | ∅ | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun | S | S | ∅ | S | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

**S in [0, N-1]:**
**This Grammar**
**can generate S**

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Recognition

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun | S | S | ∅ | S | *Joe* | 0 |
| | | Verb | VP | ∅ | VP | *eats* | 1 |
| | | | Noun | ∅ | NP | *pasta* | 2 |
| | | | | P | PP | *with* | 3 |
| | | | | | Noun | *sauce* | 4 |

**S in [0, N-1]:**
**This Grammar**
**can generate S**

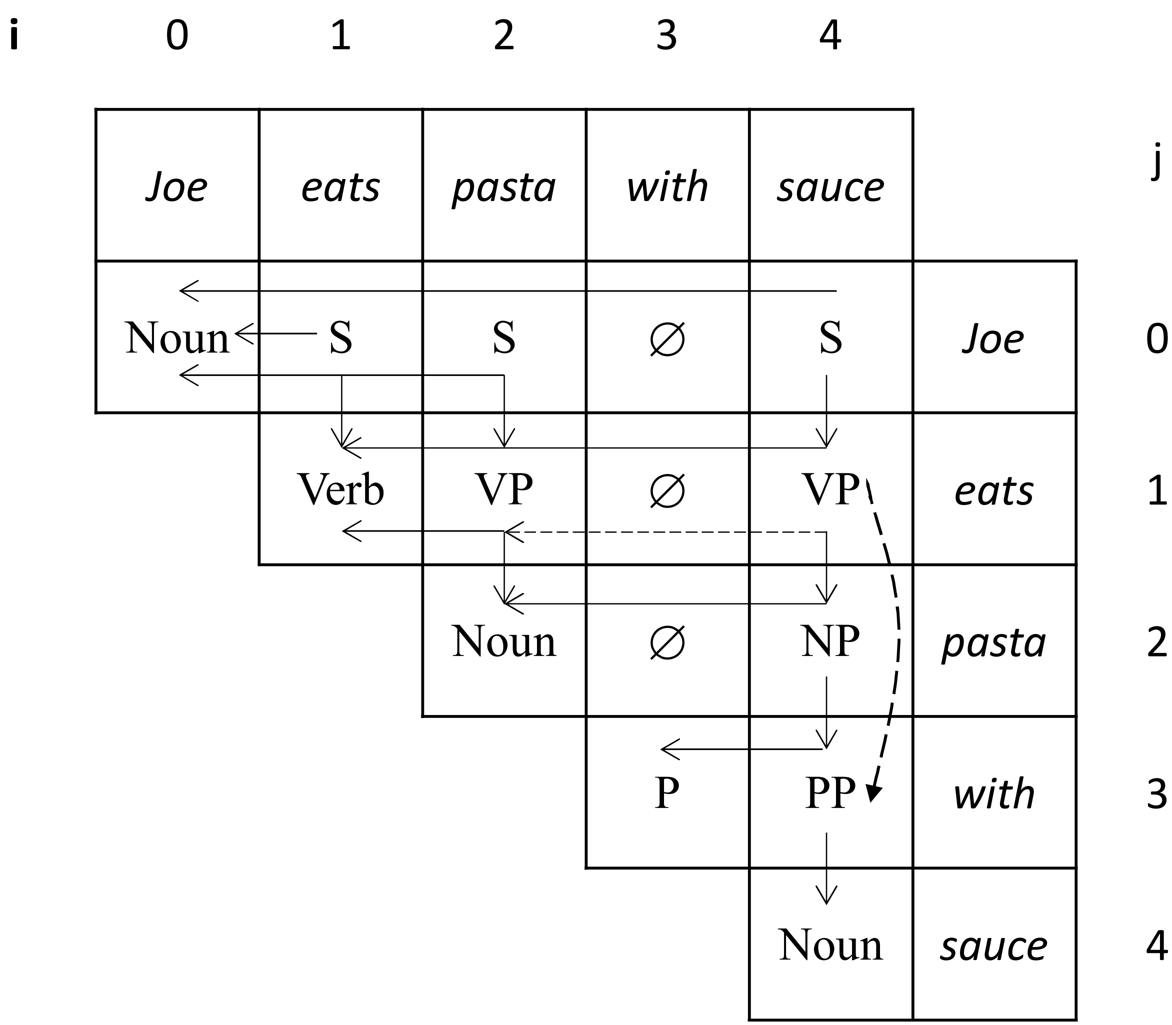
| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# CKY Parsing with Backpointers

**Input: Part Of Speech-tagged sentence S:**

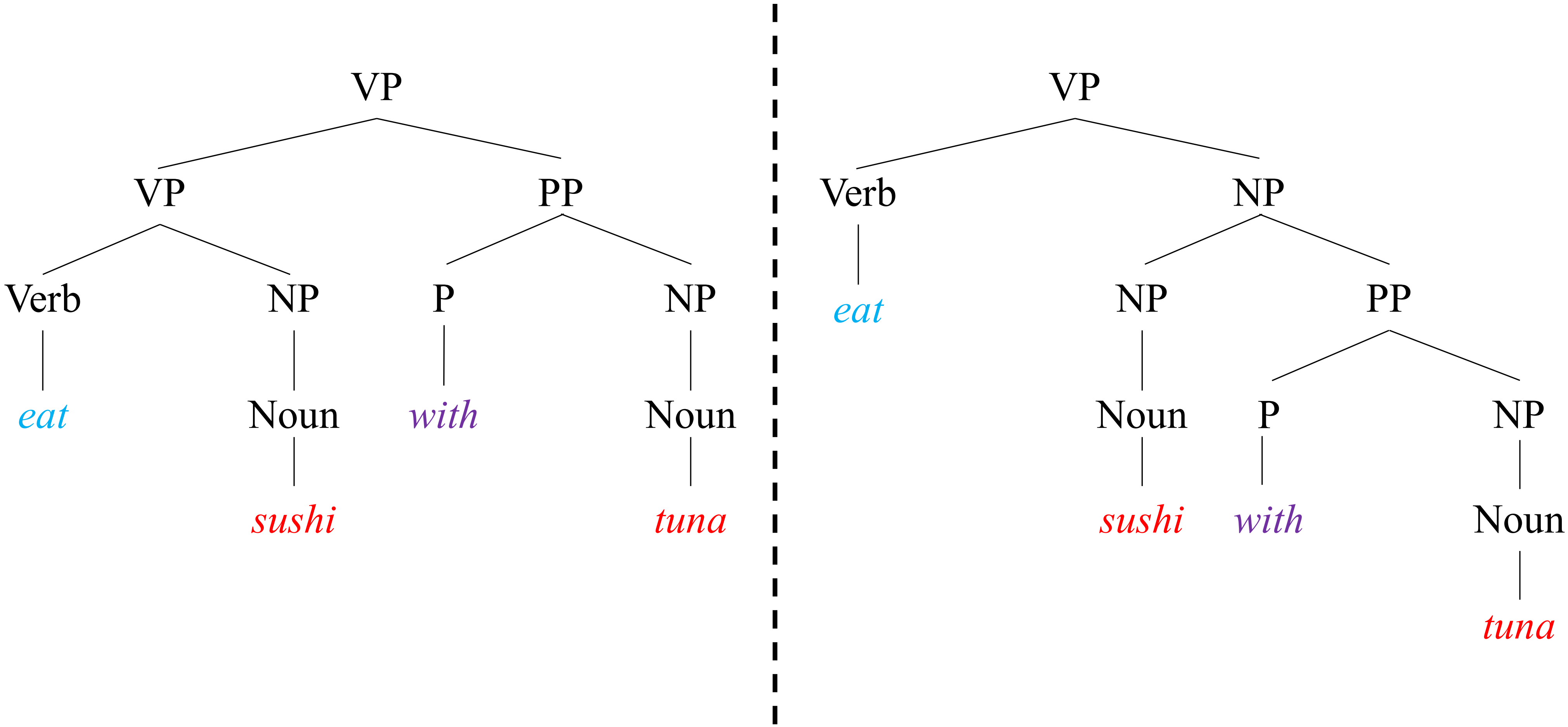
$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***




| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

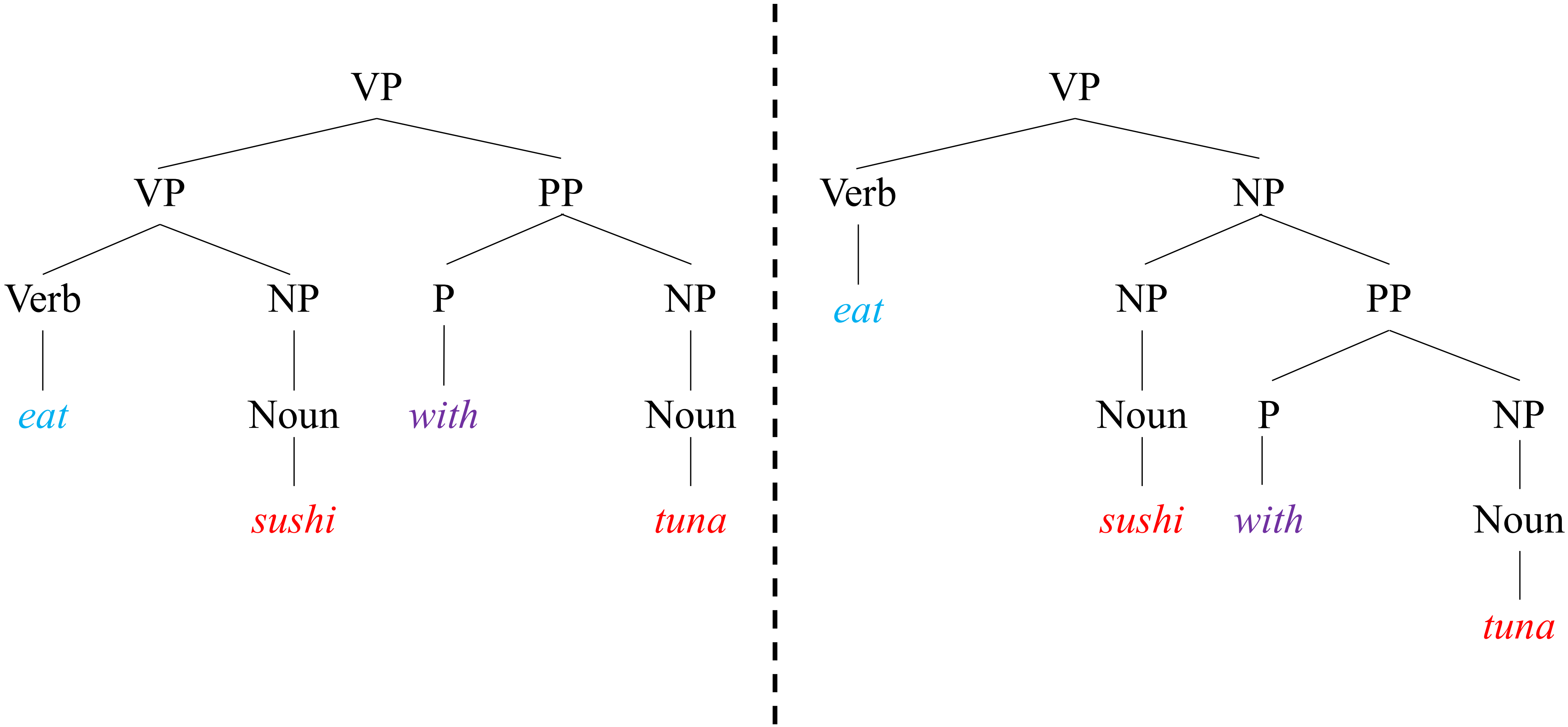
S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# Ambiguous Grammar




**What is the most likely parse tree $\tau$ for this sentence $s$?**

# Ambiguous Grammar




**What is the most likely parse tree $\tau$ for this sentence $s$?**

**We need a model of $P(\tau \mid s)$**

# Computing $\mathrm{P}(\tau \mid \mathrm{s})$ To Choose $\tau$

**Goal: find a parse tree $\tau$ such that is $\mathrm{P}(\tau \mid \mathrm{s})$ maximized**

**By Bayes Theorem:**

$$\mathrm{P}(\tau \mid \mathrm{s}) = \frac{\mathrm{P}(\tau, \mathrm{s})}{\mathrm{P}(\mathrm{s})}$$

# Computing $P(\tau \mid s)$ To Choose $\tau$

**Maximizing:**

$$P(\tau \mid s) = \frac{P(\tau, s)}{P(s)}$$

**means finding:**

$$\underset{\tau}{arg\,max}\ P(\tau \mid s) = \underset{\tau}{arg\,max}\ \frac{P(\tau, s)}{P(S)}$$

$$\underset{\tau}{arg\,max}\ P(\tau \mid s) \propto \underset{\tau}{arg\,max}\ P(\tau, s)$$

$\propto$ - proportional to

# Computing $\mathrm{P}(\tau \mid \mathrm{s})$ To Choose $\tau$

**Now:**

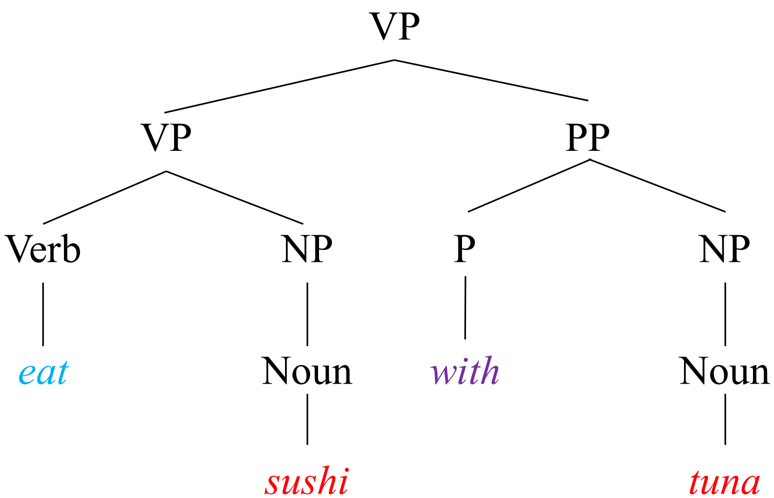
$$\underset{\tau}{arg\,max}\ \mathrm{P}(\tau, \mathrm{s}) = \underset{\tau}{arg\,max}\ \mathrm{P}(\tau)$$

$$\text{if } \mathrm{s} = \mathrm{yield}(\tau)$$

**where:**

- **yield of a tree $\tau$ is the string of terminal symbols that can be read off the leaf nodes**

# Yield of Tree τ

yield( 

) = eat sushi with tuna

# Computing $\mathrm{P}(\tau)$: The idea

**Let $\mathrm{T}$ be the (infinite) set of all trees in the language:**

$$L = \{s \in \Sigma^* | \; \exists \tau \in T : \text{yield}(\tau) = s\}$$

**We need to define $\mathrm{P}(\tau)$ such that:**

$$\forall \tau \in T : \quad 0 \leq P(\tau) \leq 1$$

$$\Sigma_{\tau \in T} P(\tau) = 1$$

# Probabilistic Context Free Grammar

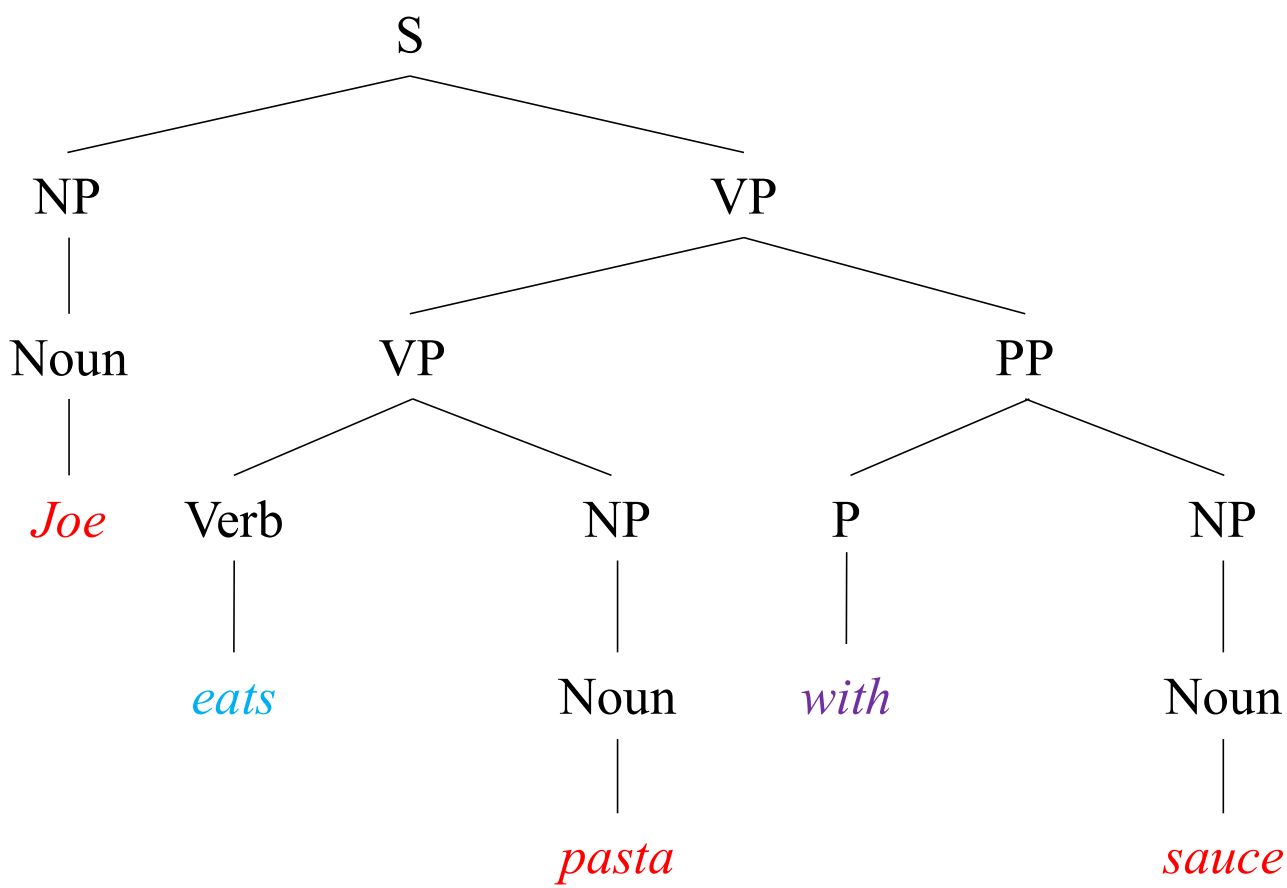
**A probabilistic context-free grammar $G$ is defined by:**

- $N$: a set of **non-terminal symbols** (variables)
- $\Sigma$: a set of **terminal symbols** (disjoint from $N$)
- $R$: a set of **rules or productions** of the form $A \rightarrow \beta$, where:
  - $A$: a non-terminal symbol
  - $\beta$: a string of symbols from the infinite set of strings $(\Sigma \cup N)^*$
- $S$: a designated **start symbol**
- $P$: the set of **probabilities** on productions / rules

# Parse Tree: Example

**Parse tree τ (representing a sequence of expansions = derivation) for sentence:**

*Joe eats pasta with sauce*



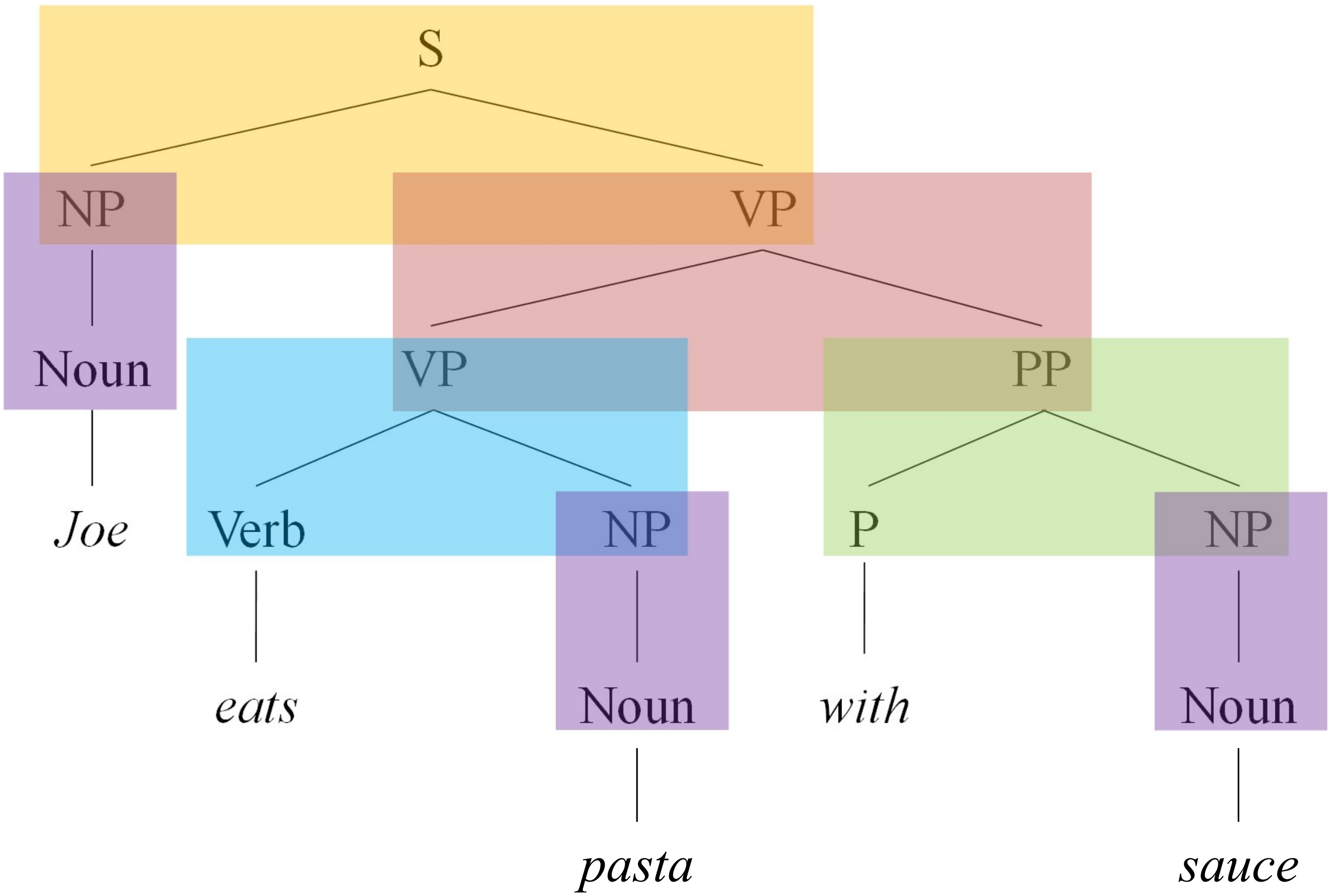

| **Rule \| Production** |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# Parse Tree: Example

**Parse tree $\tau$ (representing a sequence of expansions = derivation) for sentence:**

*Joe eats pasta with sauce*



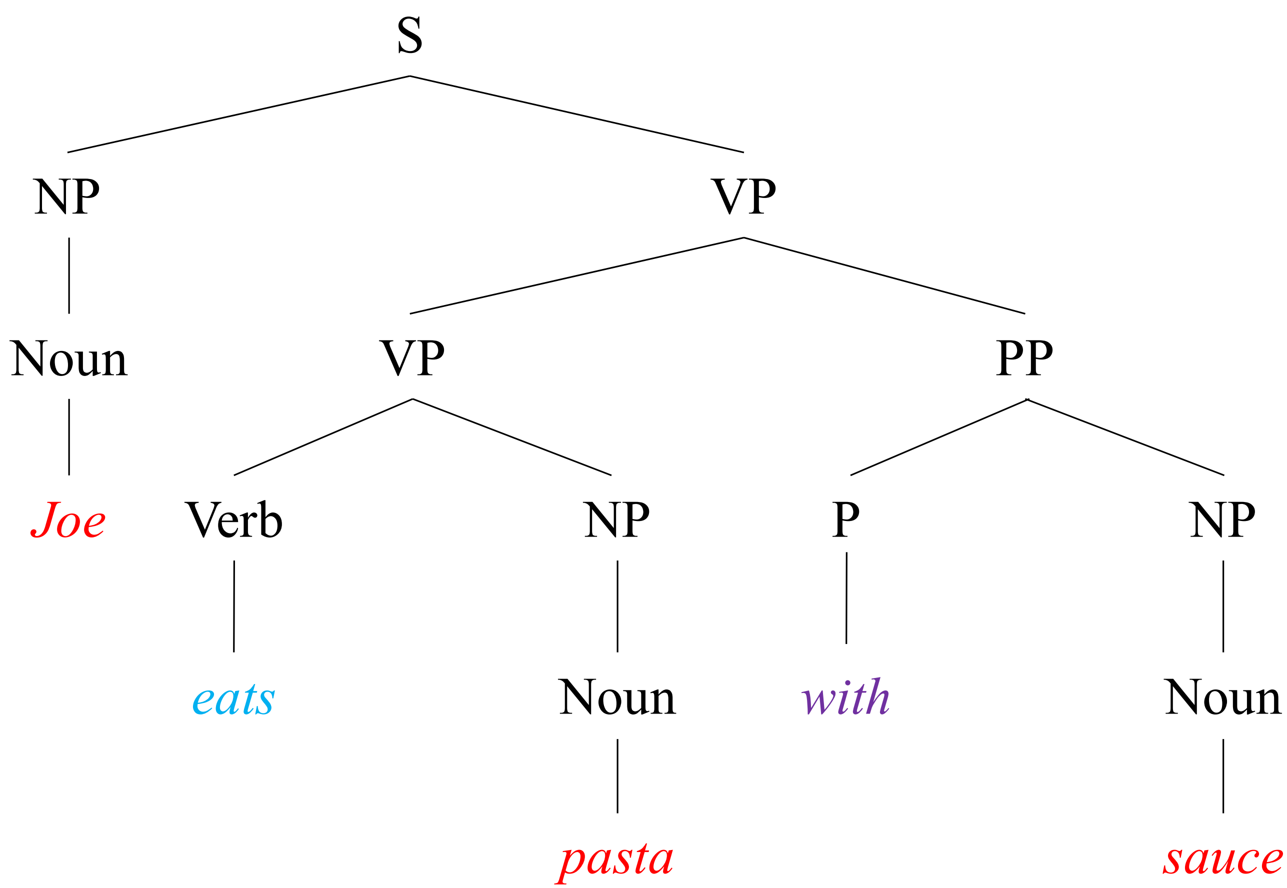

| Rule \| Production |
|---|
| S → NP VP |
| NP → Noun |
| NP → Det Noun |
| NP → NP PP |
| VP → Verb |
| VP → Verb NP |
| VP → VP PP |
| PP → P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# Parse Tree: Example

**Parse tree $\tau$ (representing a sequence of expansions = derivation) for sentence:**

*Joe eats pasta with sauce*




| Rule \| Production |
|---|
| S $\rightarrow$ NP VP |
| NP $\rightarrow$ Noun |
| NP $\rightarrow$ Det Noun |
| NP $\rightarrow$ NP PP |
| VP $\rightarrow$ Verb |
| VP $\rightarrow$ Verb NP |
| VP $\rightarrow$ VP PP |
| PP $\rightarrow$ P NP |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

$$P(\tau) = ???$$

# Probabilistic Context-Free Grammar

**For every non-terminal $X$, define a probability distribution $P(X \rightarrow \alpha \mid X)$ over all rules with the same non-terminal symbol $X$.**

| Rule \| Production | $P(X \rightarrow \alpha \mid X)$ |
|---|---|
| $S \rightarrow NP\ VP$ | 1.0 |
| $NP \rightarrow Noun$ | 0.2 |
| $NP \rightarrow Det\ Noun$ | 0.4 |
| $NP \rightarrow NP\ PP$ | 0.4 |
| $VP \rightarrow Verb$ | 0.4 |
| $VP \rightarrow Verb\ NP$ | 0.3 |
| $VP \rightarrow VP\ PP$ | 0.3 |
| $PP \rightarrow P\ NP$ | 1.0 |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# Probabilistic Context-Free Grammar

**For every non-terminal $X$, define a probability distribution $P(X \rightarrow \alpha \mid X)$ over all rules with the same non-terminal symbol $X$.**

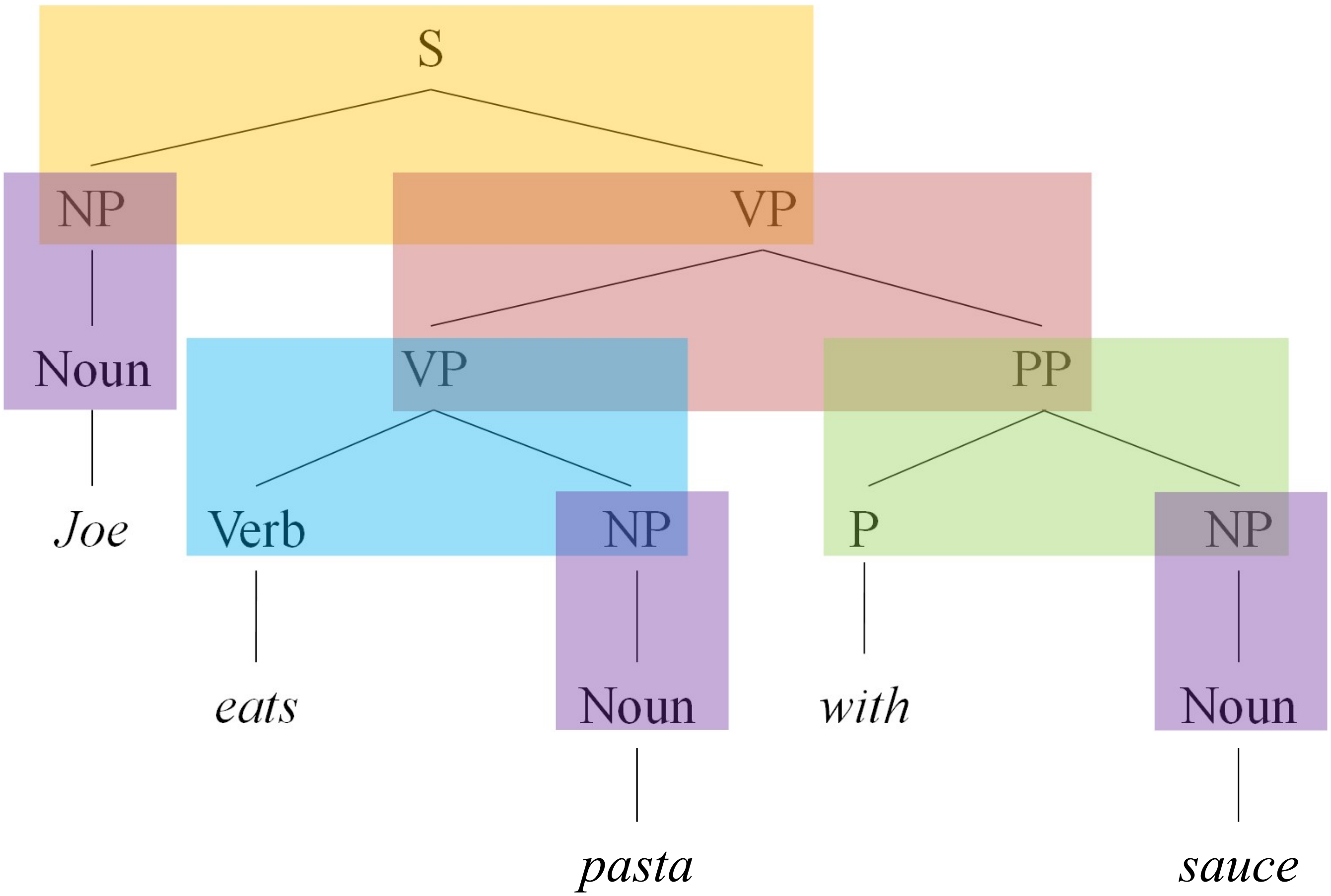
| Rule \| Production | $P(X \rightarrow \alpha \mid X)$ |
|---|---|
| $S \rightarrow NP\ VP$ | 1.0 |
| $NP \rightarrow Noun$ | 0.2 |
| $NP \rightarrow Det\ Noun$ | 0.4 |
| $NP \rightarrow NP\ PP$ | 0.4 |
| $VP \rightarrow Verb$ | 0.4 |
| $VP \rightarrow Verb\ NP$ | 0.3 |
| $VP \rightarrow VP\ PP$ | 0.3 |
| $PP \rightarrow P\ NP$ | 1.0 |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# Parse Tree: Probability

**Parse tree τ (representing a sequence of expansions = derivation) for sentence:**

*Joe eats pasta with sauce*




| Rule \| Production | P(Rule) |
|---|---|
| S → NP VP | 1.0 |
| NP → Noun | 0.2 |
| NP → Det Noun | 0.4 |
| NP → NP PP | 0.4 |
| VP → Verb | 0.4 |
| VP → Verb NP | 0.3 |
| VP → VP PP | 0.3 |
| PP → P NP | 1.0 |

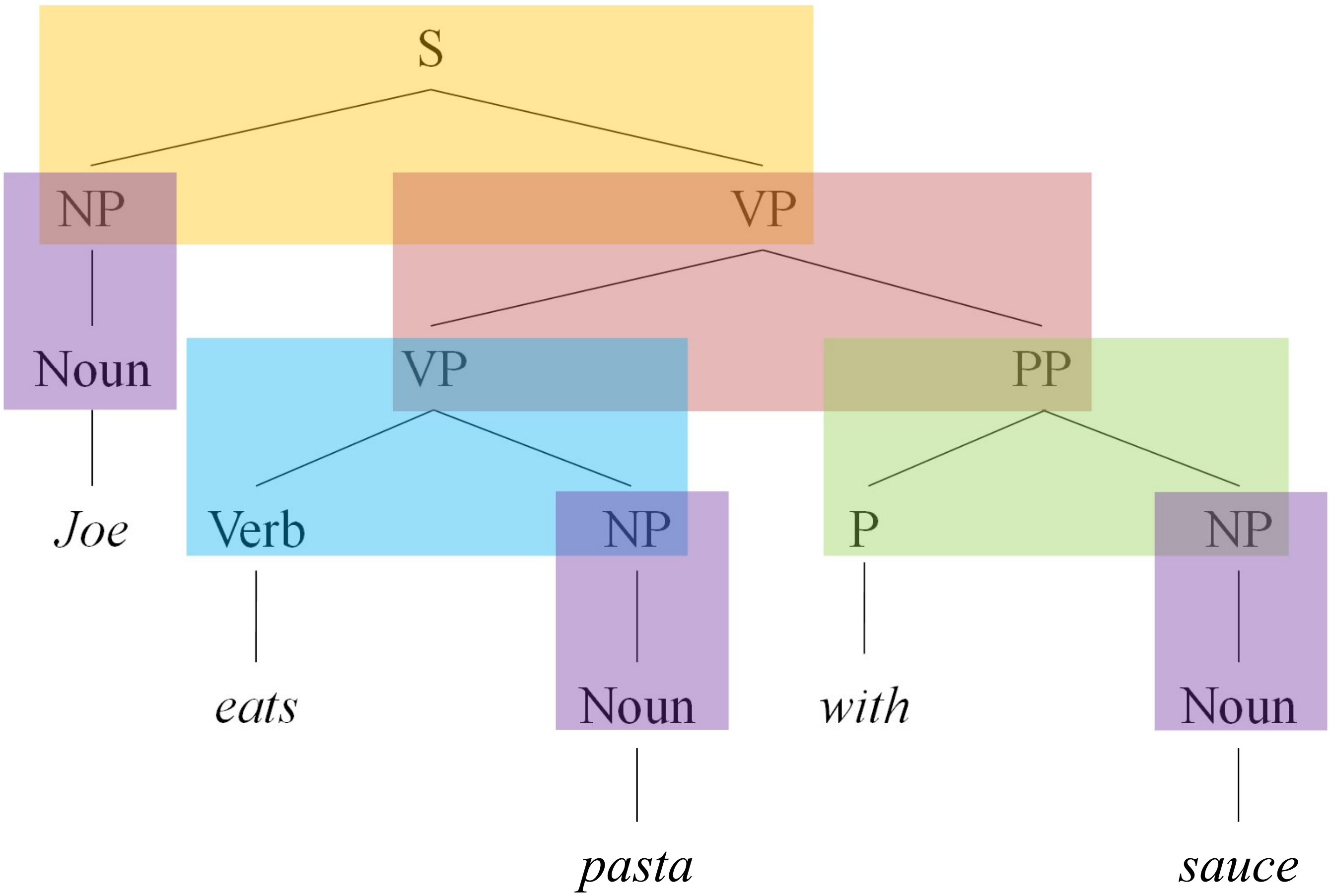
S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

$$P(\tau) = ???$$

# Parse Tree: Probability

**Parse tree $\tau$ (representing a sequence of expansions = derivation) for sentence:**

*Joe eats pasta with sauce*




| Rule \| Production | P(Rule) |
|---|---|
| S → NP VP | 1.0 |
| NP → Noun | 0.2 |
| NP → Det Noun | 0.4 |
| NP → NP PP | 0.4 |
| VP → Verb | 0.4 |
| VP → Verb NP | 0.3 |
| VP → VP PP | 0.3 |
| PP → P NP | 1.0 |

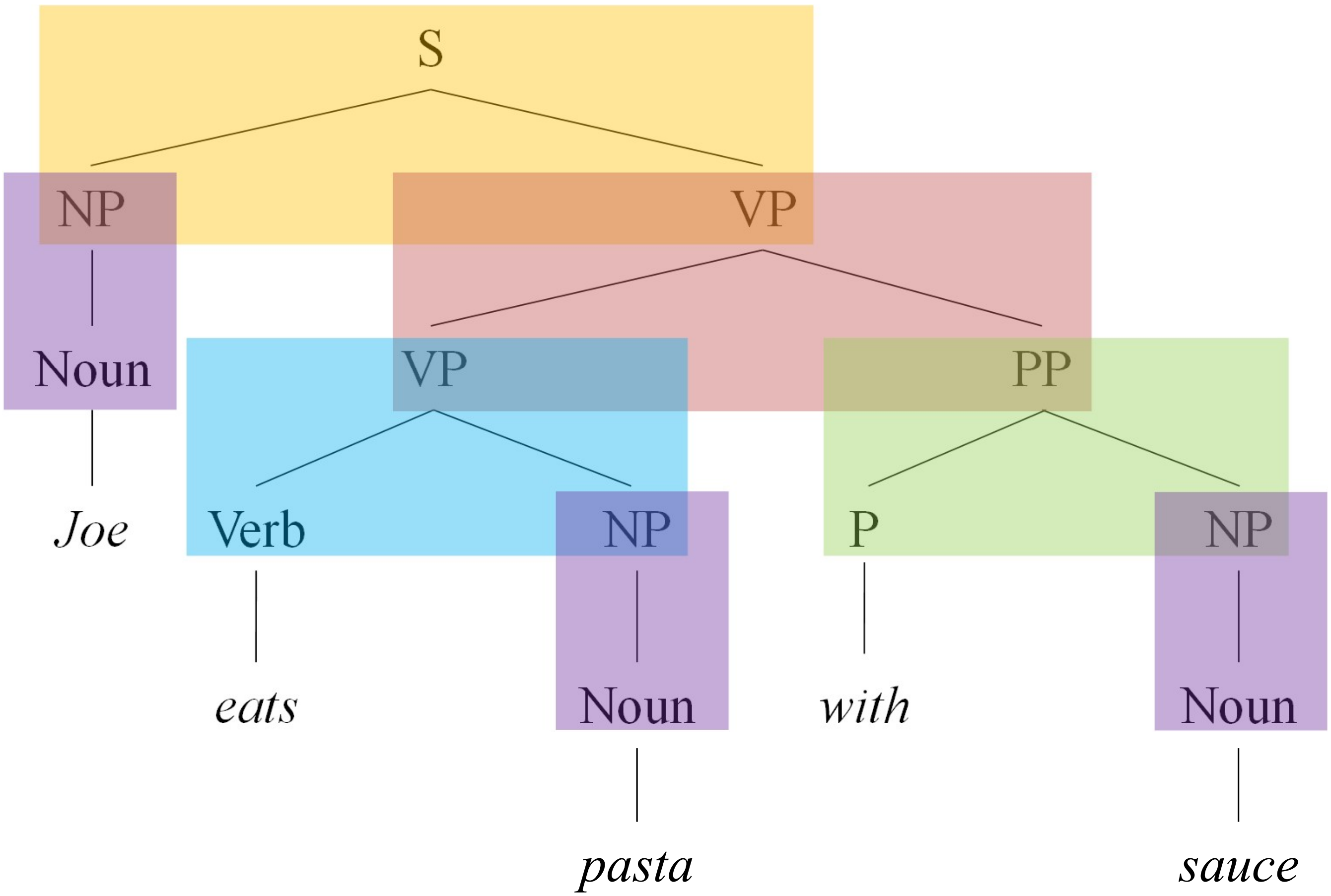
S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

$$P(\tau) = \prod P(\text{Rule})$$

# Parse Tree: Probability

**Parse tree τ (representing a sequence of expansions = derivation) for sentence:**

*Joe eats pasta with sauce*




| Rule \| Production | P(Rule) |
|---|---|
| S → NP VP | 1.0 |
| NP → Noun | 0.2 |
| NP → Det Noun | 0.4 |
| NP → NP PP | 0.4 |
| VP → Verb | 0.4 |
| VP → Verb NP | 0.3 |
| VP → VP PP | 0.3 |
| PP → P NP | 1.0 |

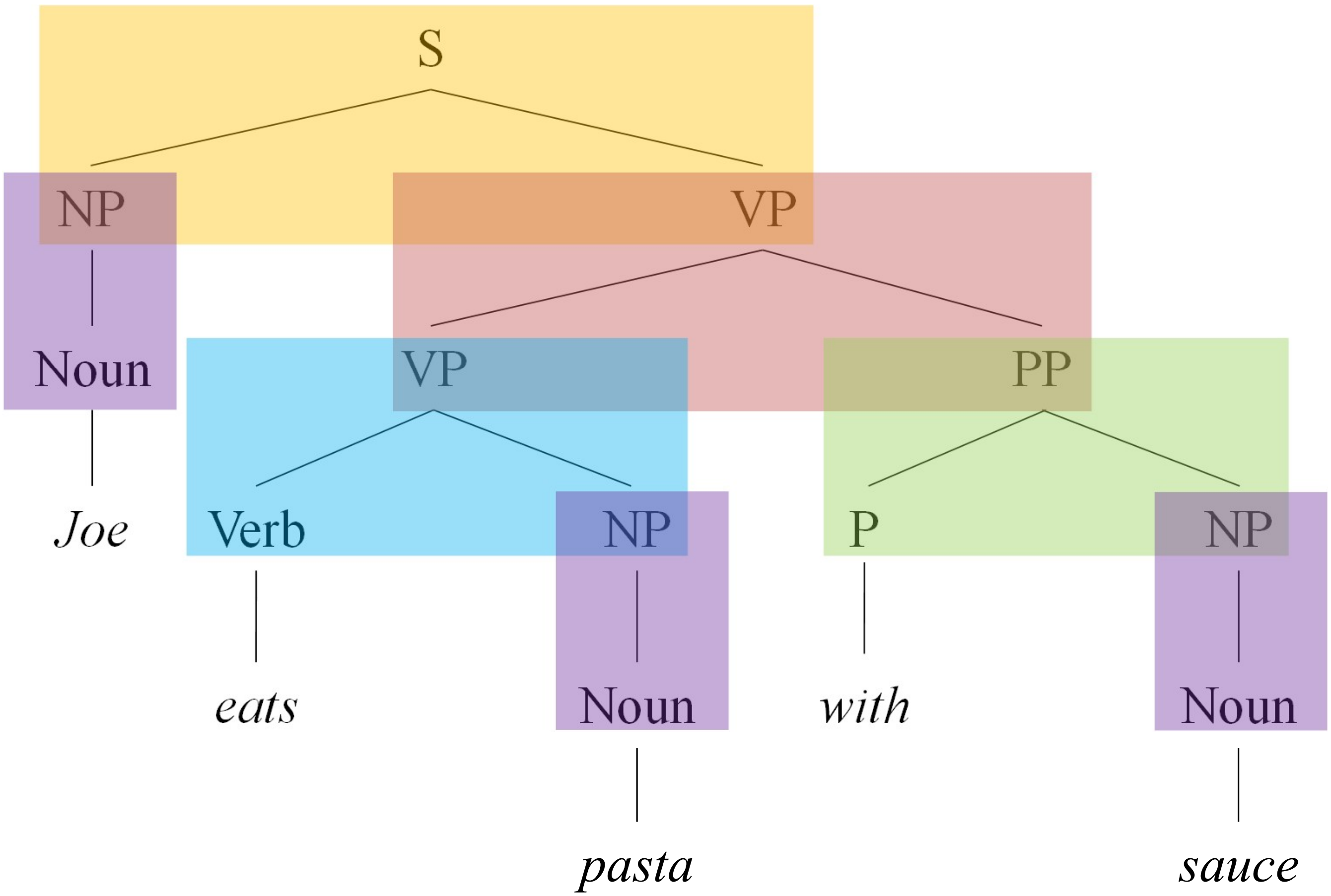
S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

$$P(\tau) = 1.0 * 0.2 * 0.3 * 0.3 * 0.2 * 1.0 * 0.2$$

# Parse Tree: Probability

**Parse tree τ (representing a sequence of expansions = derivation) for sentence:**

*Joe eats pasta with sauce*




| Rule \| Production | P(Rule) |
|---|---|
| S → NP VP | 1.0 |
| NP → Noun | 0.2 |
| NP → Det Noun | 0.4 |
| NP → NP PP | 0.4 |
| VP → Verb | 0.4 |
| VP → Verb NP | 0.3 |
| VP → VP PP | 0.3 |
| PP → P NP | 1.0 |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

$$P(\tau) = 1.0 * 0.2 * 0.3 * 0.3 * 0.2 * 1.0 * 0.2$$

# Probabilistic CKY

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun P() | S P() | S P() | ∅ P() | S P() | *Joe* | 0 |
| | | Verb P() | VP P() | ∅ P() | VP P() | *eats* | 1 |
| | | | Noun P() | ∅ P() | NP P() | *pasta* | 2 |
| | | | | P P() | PP P() | *with* | 3 |
| | | | | | Noun P() | *sauce* | 4 |

| Rule \| Production | P(Rule) |
|---|---|
| S → NP VP | 1.0 |
| NP → Noun | 0.2 |
| NP → Det Noun | 0.4 |
| NP → NP PP | 0.4 |
| VP → Verb | 0.4 |
| VP → Verb NP | 0.3 |
| VP → VP PP | 0.3 |
| PP → P NP | 1.0 |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# Probabilistic CKY

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun<br>P() | S<br>P() | S<br>P() | ∅<br>P() | S<br>P() | *Joe* | 0 |
| | | Verb<br>P() | VP<br>P() | ∅<br>P() | VP<br>P() | *eats* | 1 |
| | | | Noun<br>P() | ∅<br>P() | NP<br>P() | *pasta* | 2 |
| | | | | P<br>P() | PP<br>P() | *with* | 3 |
| | | | | | Noun<br>P() | *sauce* | 4 |

| Rule \| Production | P(Rule) |
|---|---|
| S → NP VP | 1.0 |
| NP → Noun | 0.2 |
| NP → Det Noun | 0.4 |
| NP → NP PP | 0.4 |
| VP → Verb | 0.4 |
| VP → Verb NP | 0.3 |
| VP → VP PP | 0.3 |
| PP → P NP | 1.0 |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

# Probabilistic CKY

**Input: Part Of Speech-tagged sentence S:**

$_0$ *Joe*|***N*** $_1$ *eats*|***V*** $_2$ *pasta*|***N*** $_3$ *with*|***P*** $_4$ *sauce*|***N***

| i | 0 | 1 | 2 | 3 | 4 | | j |
|---|---|---|---|---|---|---|---|
| | *Joe* | *eats* | *pasta* | *with* | *sauce* | | |
| | Noun P() | S P() | S P() | ∅ P() | S P() | *Joe* | 0 |
| | | Verb P() | VP P() | ∅ P() | VP P() | *eats* | 1 |
| | | | Noun P() | ∅ P() | NP P() | *pasta* | 2 |
| | | | | P P() | PP P() | *with* | 3 |
| | | | | | Noun P() | *sauce* | 4 |

| Rule \| Production | P(Rule) |
|---|---|
| S → NP VP | 1.0 |
| NP → Noun | 0.2 |
| NP → Det Noun | 0.4 |
| NP → NP PP | 0.4 |
| VP → Verb | 0.4 |
| VP → Verb NP | 0.3 |
| VP → VP PP | 0.3 |
| PP → P NP | 1.0 |

S-start symbol | NP-noun phrase | VP- verb phrase | conj-conjunction | Det-determiner | PP-prepositional phrase | P-preposition

**Multiple possible trees!**
**Multiple probabilites**
**Pick max!**