PRACTICAL 9

CYK ALGORITHM REVISITED

Presented by Qing Lyu / 吕晴

CONVERSION TO CHOMSKY NORMAL FORM

- Requirement: grammars used with the CYK algorithm must be in Chomsky Normal Form (CNF), i.e. each rule is of the form:

```
A → B C or A → w
```

CONVERSION TO CHOMSKY NORMAL FORM

Restriction: `A → B C or A → w`

Original Grammar

$S \rightarrow VP$	$VP \rightarrow V\ NP\ PP$
$NP \rightarrow Prn$	$PP \rightarrow P\ NP$
$NP \rightarrow Det\ Nom$	$Det \rightarrow the \mid a$
$Nom \rightarrow N$	$N \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$	$V \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V$	$Prn \rightarrow I \mid she \mid me$
$VP \rightarrow V\ NP$	$P \rightarrow from \mid for \mid to$

CONVERSION TO CHOMSKY NORMAL FORM

Restriction: `A → B C or A → w`

Original Grammar

$S \rightarrow VP$	$VP \rightarrow V\ NP\ PP$
$NP \rightarrow Prn$	$PP \rightarrow P\ NP$
$NP \rightarrow Det\ Nom$	$Det \rightarrow the \mid a$
$Nom \rightarrow N$	$N \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$	$V \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V$	$Prn \rightarrow I \mid she \mid me$
$VP \rightarrow V\ NP$	$P \rightarrow from \mid for \mid to$

2 kinds of offending rules:

- A → B
- A → B C D

(also: A → w B)

CONVERSION TO CHOMSKY NORMAL FORM

- For the 1st kind , A → B (unit productions), we can rewrite the right-hand side of the original rules with the right-hand side of all the non-unit production rules that they ultimately lead to:

e.g.

Original Grammar

$NP \rightarrow Prn$

$Prn \rightarrow I \mid she \mid me$

CNF form

$NP \rightarrow I \mid she \mid me$

$Prn \rightarrow I \mid she \mid me$

- flattening of the grammar

CONVERSION TO CHOMSKY NORMAL FORM

- For the 2nd kind, A → B C D (right-hand side length > 2), we can replace a pair of non-terminals with a new non-terminal and introduce a new production rule:

e.g. Original Grammar CNF form

$VP \rightarrow V\ NP\ PP$

$VP \rightarrow X1\ PP$

$X1 \rightarrow\ V\ NP$

- In the case of longer right-hand sides, we simply iterate this process until the offending rule has been replaced by rules of length 2.

CONVERSION TO CHOMSKY NORMAL FORM

- 3rd kind?

e.g.

Original Grammar		CNF form
INF-VP → to VP		???

CONVERSION TO CHOMSKY NORMAL FORM

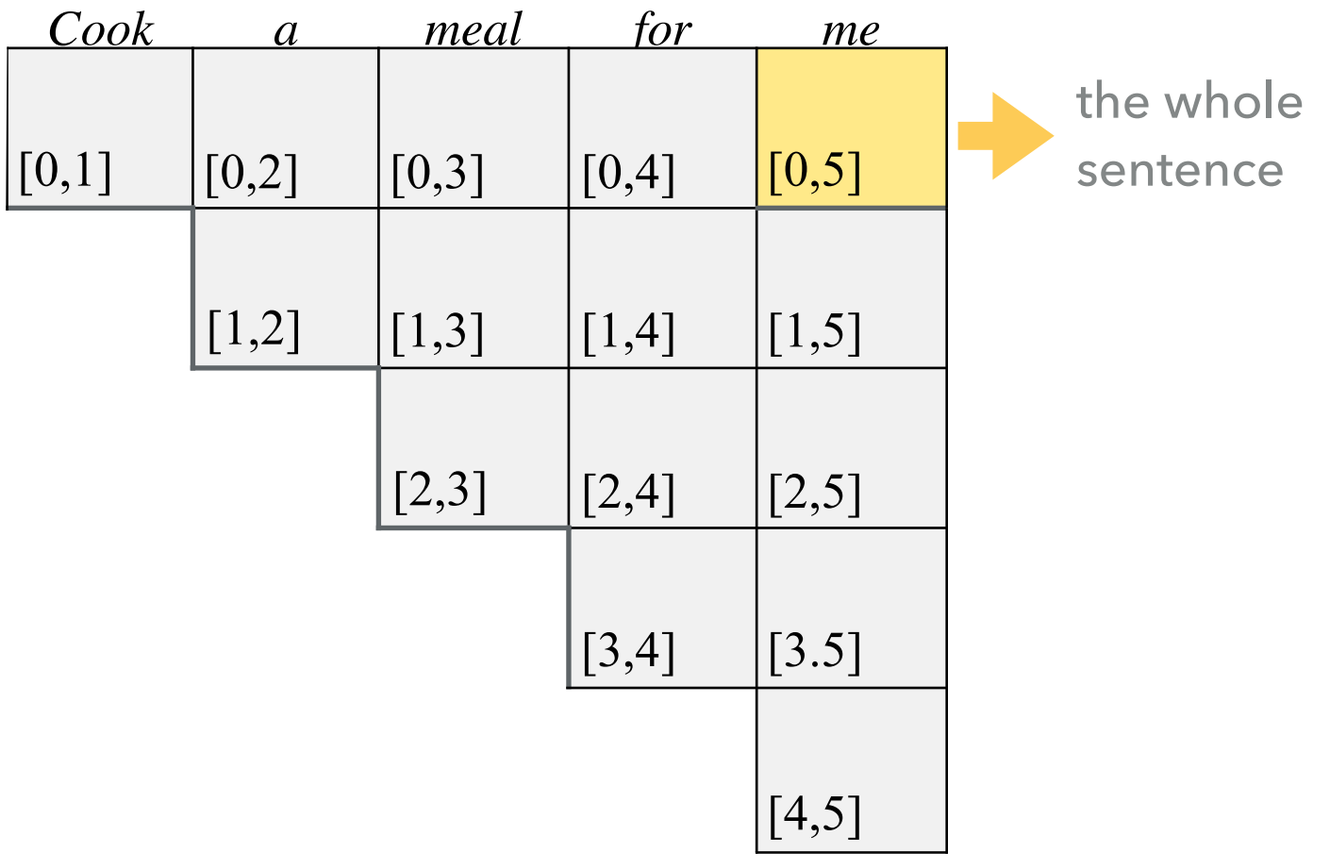
$S \rightarrow VP$	$???$
	$???$
	$???$
$NP \rightarrow Prn$	$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$	$NP \rightarrow Det\ Nom$
$Nom \rightarrow N$	$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$	$Nom \rightarrow Nom\ PP$
$VP \rightarrow V$	$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$	$VP \rightarrow V\ NP$
$VP \rightarrow V\ NP\ PP$	$VP \rightarrow X1\ PP$
	$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$	$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$	$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$	$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$	$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$	$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$	$P \rightarrow from \mid for \mid to$

CONVERSION TO CHOMSKY NORMAL FORM

$S \rightarrow VP$	$S \rightarrow book \mid cook \mid prefer$
	$S \rightarrow V\ NP$
	$S \rightarrow X1\ PP$
$NP \rightarrow Prn$	$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$	$NP \rightarrow Det\ Nom$
$Nom \rightarrow N$	$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$	$Nom \rightarrow Nom\ PP$
$VP \rightarrow V$	$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$	$VP \rightarrow V\ NP$
$VP \rightarrow V\ NP\ PP$	$VP \rightarrow X1\ PP$
	$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$	$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$	$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$	$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$	$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$	$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$	$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- For a sentence of length n, we will work with the upper-triangular portion of an (n + 1) × (n + 1) matrix
- e.g.

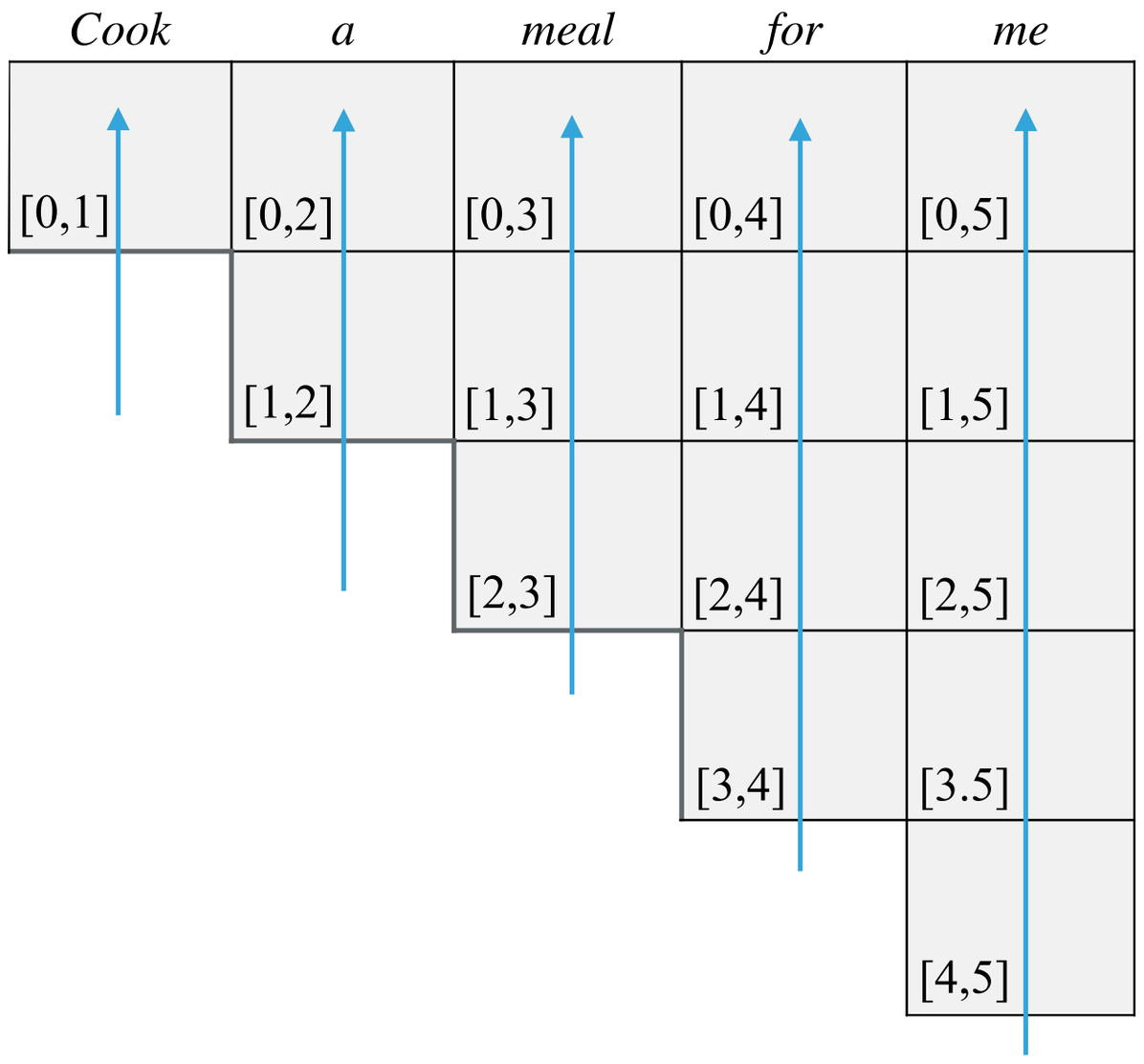


$_0$ *Cook* $_1$ *a* $_2$ *meal* $_3$ *for* $_4$ *me* $_5$

- Each cell [i, j] in this matrix contains the set of non-terminals that represent all the constituents that span positions i through j of the input

CYK ALGORITHM

- For each cell [i, j], there must be a position in the input, k, where the constituent represented by [i, j] can be split into two parts such that i < k < j.



CYK ALGORITHM

- The superdiagonal row in the matrix contains the parts of speech (actually, all possible pre-terminals) for each input word in the input
- e.g.

Cook	*a*	*meal*	*for*	*me*
[0,1]	[0,2]	[0,3]	[0,4]	[0,5]
	[1,2]	[1,3]	[1,4]	[1,5]
		[2,3]	[2,4]	[2,5]
			[3,4]	[3.5]
				[4,5]

Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The superdiagonal row in the matrix contains the parts of speech (actually, all possible pre-terminals) for each input word in the input
- e.g.

Cook	*a*	*meal*	*for*	*me*
S, Nom, VP, N, V [0,1]	[0,2]	[0,3]	[0,4]	[0,5]
	Det [1,2]	[1,3]	[1,4]	[1,5]
		N, Nom [2,3]	[2,4]	[2,5]
			P [3,4]	[3.5]
				NP, Prn [4,5]

Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The subsequent diagonals above that superdiagonal contain constituents that cover all the spans of increasing length in the input

Cook	*a*	*meal*	*for*	*me*
S, Nom, VP, N, V [0,1]	[0,2]	[0,3]	[0,4]	[0,5]
	Det [1,2]	[1,3]	[1,4]	[1,5]
		N, Nom [2,3]	[2,4]	[2,5]
			P [3,4]	[3.5]
				NP, Prn [4,5]

Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The subsequent diagonals above that superdiagonal contain constituents that cover all the spans of increasing length in the input

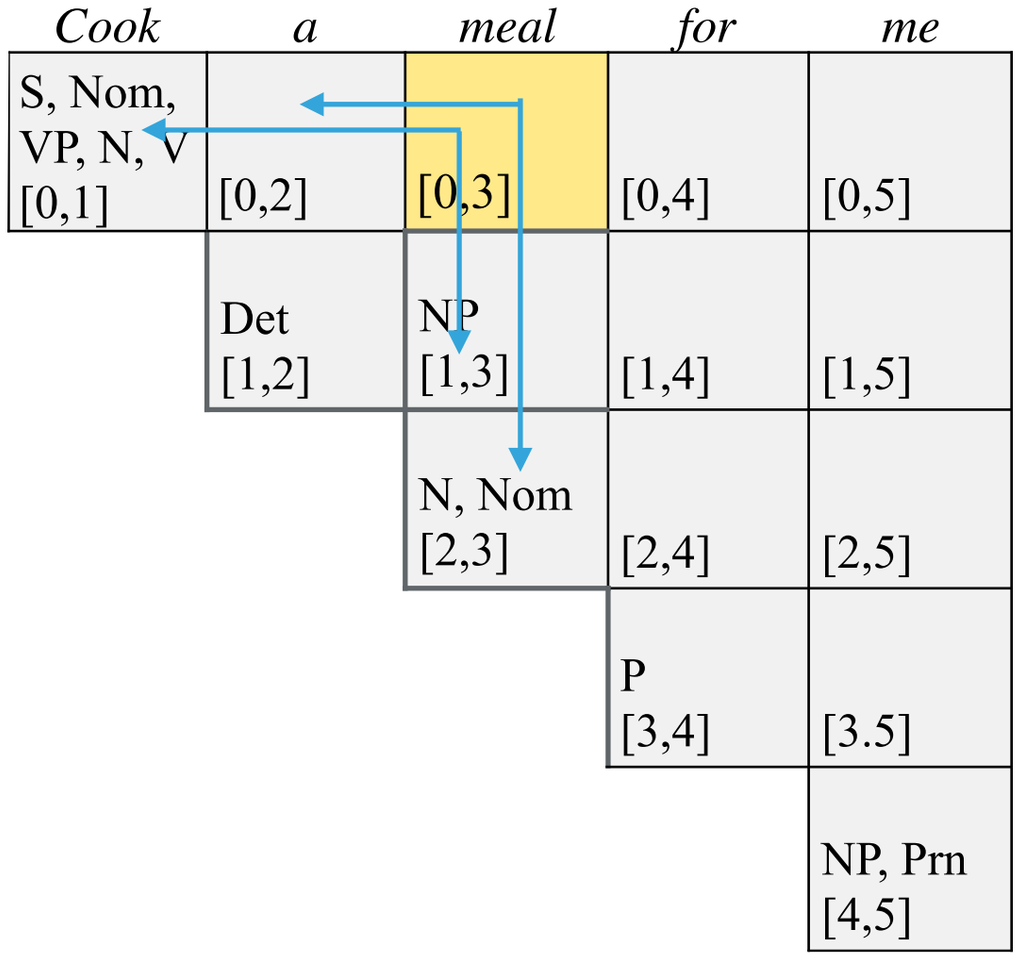
Cook	*a*	*meal*	*for*	*me*
S, Nom, VP, N, V [0,1]	[0,2]	[0,3]	[0,4]	[0,5]
	Det [1,2]	[1,3]	[1,4]	[1,5]
		N, Nom [2,3]	[2,4]	[2,5]
			P [3,4]	[3.5]
				NP, Prn [4,5]

Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The subsequent diagonals above that superdiagonal contain constituents that cover all the spans of increasing length in the input



Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The subsequent diagonals above that superdiagonal contain constituents that cover all the spans of increasing length in the input

Cook	*a*	*meal*	*for*	*me*
S, Nom, VP, N, V [0,1]	[0,2]	S, VP, X1 [0,3]	[0,4]	[0,5]
	Det [1,2]	NP [1,3]	[1,4]	[1,5]
		N, Nom [2,3]	[2,4]	[2,5]
			P [3,4]	[3.5]
				NP, Prn [4,5]

Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The subsequent diagonals above that superdiagonal contain constituents that cover all the spans of increasing length in the input

Cook	*a*	*meal*	*for*	*me*
S, Nom, VP, N, V [0,1]	[0,2]	S, VP, X1 [0,3]	[0,4]	[0,5]
	Det [1,2]	NP [1,3]	[1,4]	[1,5]
		N, Nom [2,3]	[2,4]	[2,5]
			P [3,4]	PP [3.5]
				NP, Prn [4,5]

Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The subsequent diagonals above that superdiagonal contain constituents that cover all the spans of increasing length in the input

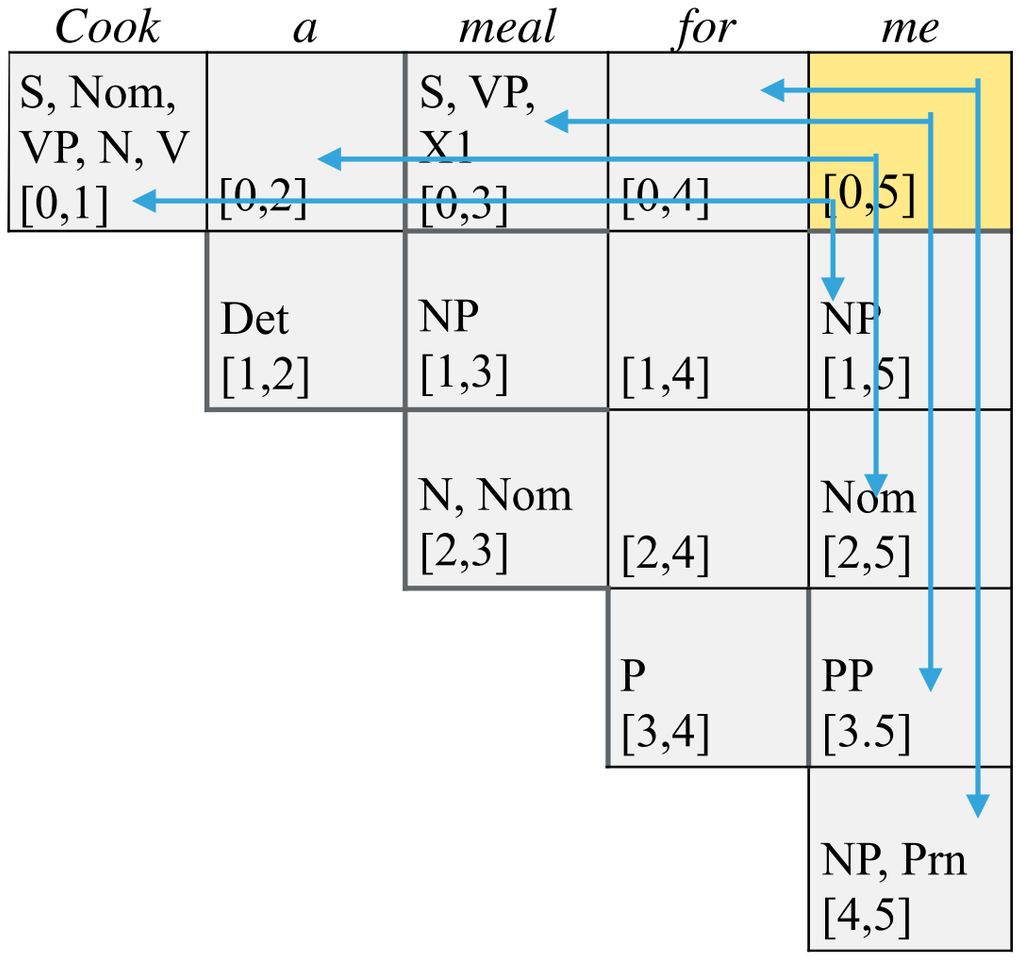
Cook	*a*	*meal*	*for*	*me*
S, Nom, VP, N, V [0,1]	[0,2]	S, VP, X1 [0,3]	[0,4]	[0,5]
	Det [1,2]	NP [1,3]	[1,4]	[1,5]
		N, Nom [2,3]	[2,4]	Nom [2,5]
			P [3,4]	PP [3.5]
				NP, Prn [4,5]

Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

- The subsequent diagonals above that superdiagonal contain constituents that cover all the spans of increasing length in the input



Our grammar

$S \rightarrow book \mid cook \mid prefer$
$S \rightarrow V\ NP$
$S \rightarrow X1\ PP$
$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Det\ Nom$
$Nom \rightarrow book \mid cook \mid meal$
$Nom \rightarrow Nom\ PP$
$VP \rightarrow book \mid cook \mid prefer$
$VP \rightarrow V\ NP$
$VP \rightarrow X1\ PP$
$X1 \rightarrow V\ NP$
$PP \rightarrow P\ NP$
$Det \rightarrow the \mid a$
$N \rightarrow book \mid cook \mid meal$
$V \rightarrow book \mid cook \mid prefer$
$Prn \rightarrow I \mid she \mid me$
$P \rightarrow from \mid for \mid to$

CYK ALGORITHM

Cook	*a*	*meal*	*for*	*me*
S, Nom, VP, N, V [0,1]	[0,2]	S, VP, X1 [0,3]	[0,4]	S, VP, X1 [0,5]
	Det [1,2]	NP [1,3]	[1,4]	NP [1,5]
		N, Nom [2,3]	[2,4]	Nom [2,5]
			P [3,4]	PP [3.5]
				NP, Prn [4,5]

- What's the result (syntactic tree) of our parsing algorithm?
  Is there any kind of ambiguities?