61A Lecture 37

Friday, November 30

What is Computer Science?

Systems

Artificial Intelligence

Graphics

Security

Networking

Programming Languages

...

Computer Vision

Planning

Robotics

Natural Language Processing

...

Machine Translation

Bonus Material

Target language corpus gives examples of well-formed sentences

I will get to it later

See you later

He will do it

Parallel corpus gives translation examples

I will do it gladly

Yo lo haré de muy buen grado

You will see later

Después lo veras

Machine translation system:

Source language

Yo lo haré después

NOVEL SENTENCE

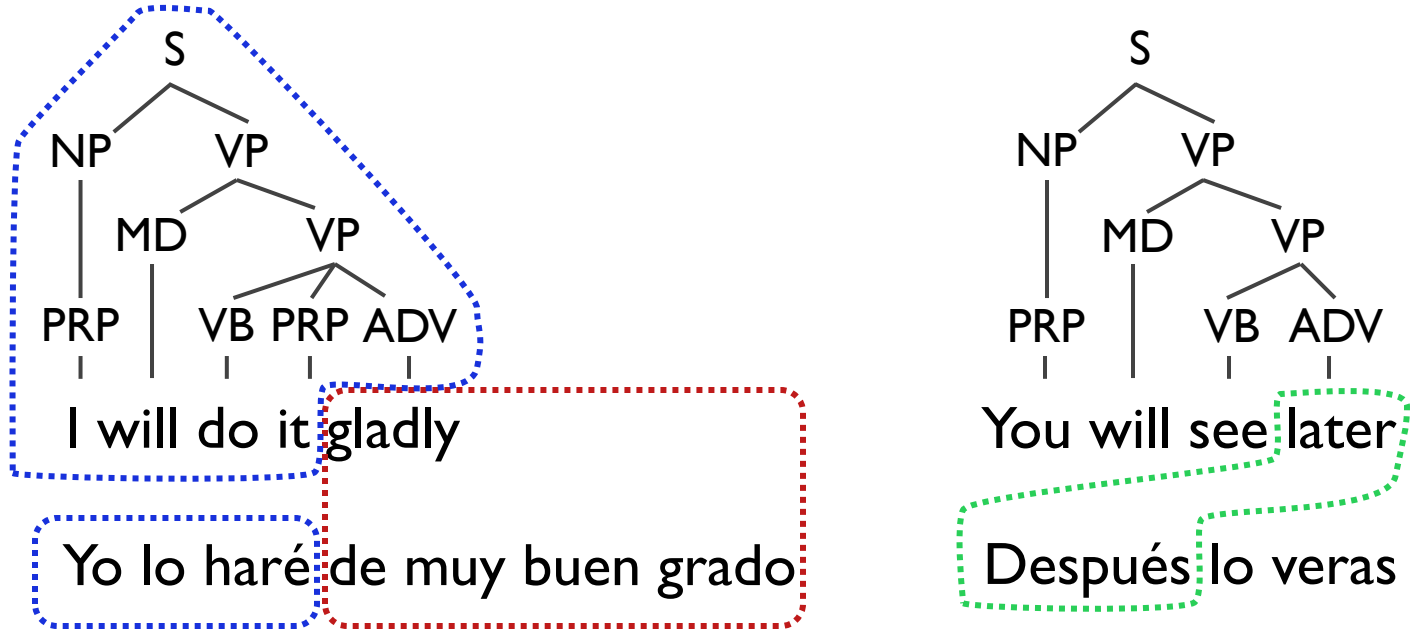
Model of translation

Target language

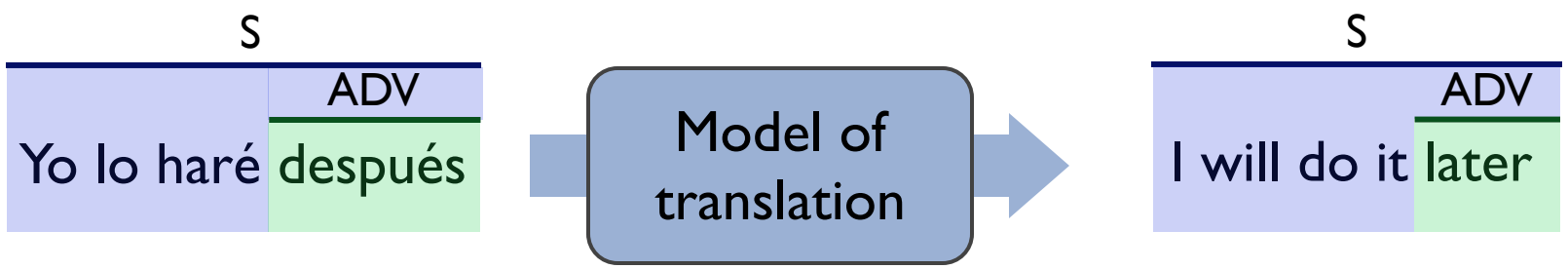
I will do it later

The Syntactic Structure of Natural Language

Parallel corpus gives translation examples



Machine translation system:



A Translation System in 20 Minutes

Context-free grammars as a model of language generation

Large-scale linguistic annotations

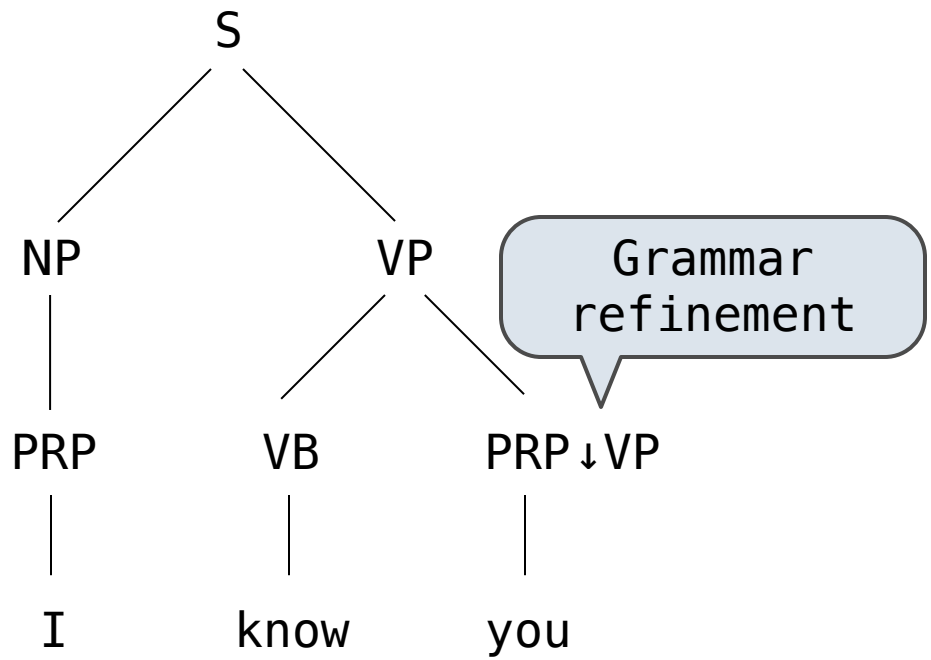
Tree transformations

English ⇨ **Yoda-speak**

Help you, I can!
Yes! Mm!

When 900 years old you reach,
look as good, you will not. Hm.

A Context-Free Grammar Describes Generation



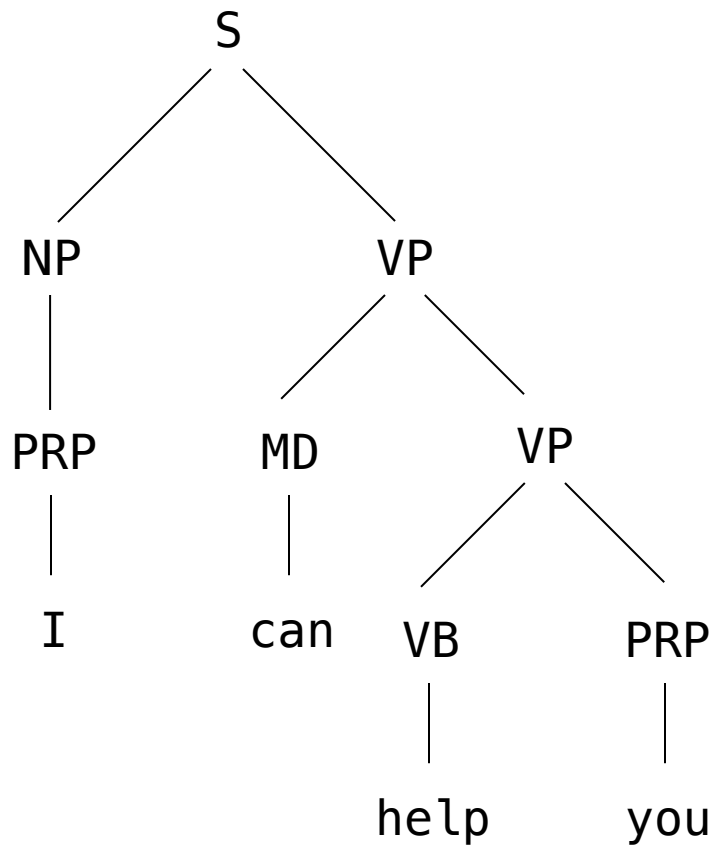
"Grammar"

```
S -> NP VP
NP -> PRP↓NP
VP -> VB
VP -> VB PRP↓VP
```

"Lexicon"

```
PRP↓NP -> I
PRP -> you
VB -> know
VB -> help
PRP↓VP -> me
```

A Context-Free Grammar Describes Generation



"Grammar"

```
S -> NP VP
NP -> PRP
VP -> VB
VP -> VB PRP
VP -> MD VP
```

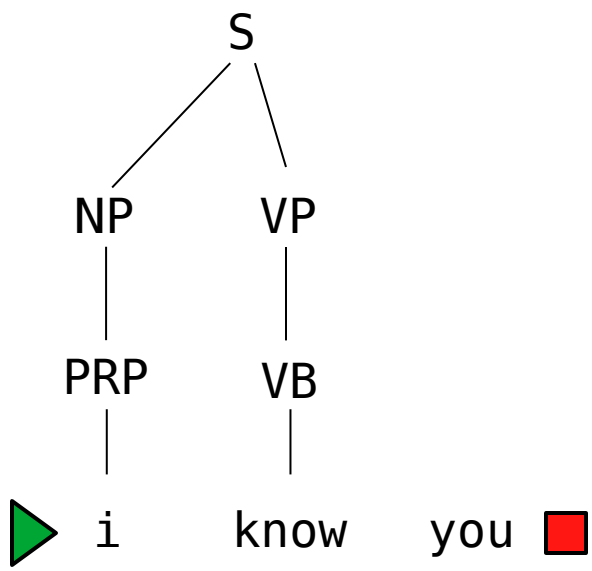
"Lexicon"

```
PRP -> I
PRP -> you
VB -> know
VB -> help
MD -> can
```

A grammar can be *learned* from data (demo)

Left-Corner Parsing Builds Trees For Sentences

```
parse_next(S, 1)
  complete(S, (PRP i), 1)
  complete(S, (NP (PRP i)), 1)
  parse_next(VP, 2)
    complete(VP, (VB know), 2)
    complete(VP, (VP (VB know)), 2)
  complete(S, (S ...), 1)
```



"Grammar"

```
S  -> NP VP
NP -> PRP
VP -> VB
VP -> VB PRP
```

Left corners

"Lexicon"

```
PRP -> I
PRP -> you
VB -> know
VB -> help
```

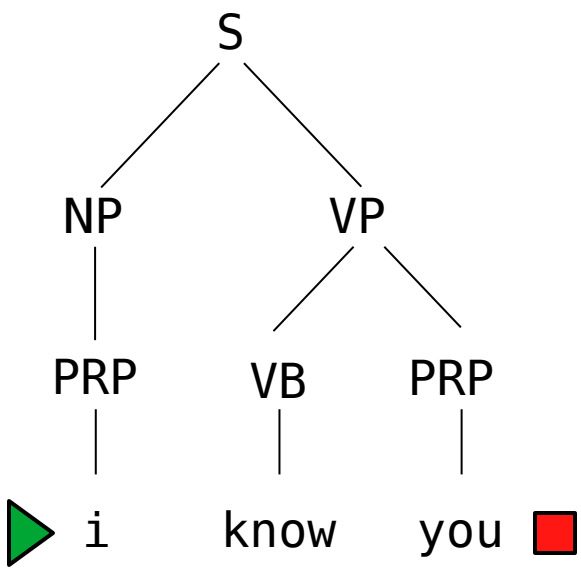
Left-Corner Parsing Builds Trees For Sentences

```
parse_next(S, 1)
  complete(S, (PRP i), 1)
  complete(S, (NP (PRP i)), 1)
  parse_next(VP, 2)
    complete(VP, (VB know), 2)
    complete(VP, (VP (VB know) (PRP you)), 2)
  complete(S, (S ...), 1)
```

"Grammar"

```
S  -> NP VP
NP -> PRP
VP -> VB
VP -> VB PRP
```

Left corners



"Lexicon"

```
PRP -> I
PRP -> you
VB -> know
VB -> help
```