CS 6501 Natural Language Processing

Probabilistic Context-Free Grammars

Yangfeng Ji

September 19, 2018

Department of Computer Science
University of Virginia

University of Virginia | ENGINEERING

Overview

1. About Group Project Proposal
2. Introduction
3. Context-Free Grammars
4. Ambiguity
5. Probabilistic CFGs
6. Penn Treebank

Based on slides from [Collins, 2017, Smith, 2018]

About Group Project Proposal

Goal of the Group Project

An opportunity to work on a larger NLP system on a topic of your choice

Proposal Outline

1. Problem definition
   - What is the problem? Input and Output?
   - Why it is interesting/challenging?
   - Is this relevant to your own research project? What is the connection?
2. Related work
   - Any existing work on this topic?
   - What is the difference?
3. Data
   - What data you will use?
   - What is the data size, in terms of the number of words/sentences/documents?

Proposal Outline (Cont.)

4. Proposed method(s)
   - What method/model/algorithm are you planning to use?
   - Does this involve some existing implementations/packages? How?
5. Evaluation plan
   - How will you evaluate your results? What evaluation metrics that you will use?
   - What are your criteria of success?

Requirements

- No more than two pages, including references
- Proposal due on Oct. 7th, 11:59PM
- Proposal template: the last NIPS submission template
  - include all the group members in the author list
  - please do not modify the template

Proposal Presentation

- Each group: 5-minute presentation and 3-minute QA
- About 5-page slides
  - one section per slide

Example Projects

In addition to the projects I suggested before

1. NLP techniques on social media
2. Conversational modeling
3. Question answering

NLP techniques on Social Media

`http://www.cs.cmu.edu/~ark/TweetNLP/`

Example projects

1. POS tagger/syntactic parser on tweets
2. Word embeddings on tweets and its application on sentiment classification

Conversational Modeling

1. Build non-goal-oriented dialogue systems
2. Build a conversational model that can generate diverse responses

Question Answering

Dataset	Multi turn	Text-based	Dialog Acts	Simple Evaluation	Unanswerable Questions	Asker Can't See Evidence
QuAC	✔	✔	✔	✔	✔	✔
CoQA (Reddy et al., 2018)	✔	✔	✗	✔	✔	✗
CSQA (Saha et al., 2018)	✔	✗	✗	✗	✔	✗
CQA (Talmor and Berant, 2018)	✔	✔	✗	✔	✗	✔
SQA (Iyyer et al., 2017)	✔	✗	✗	✔	✗	✗
NarrativeQA (Kociský et al., 2017)	✗	✔	✗	✗	✗	✔
TriviaQA (Joshi et al., 2017)	✗	✔	✗	✔	✗	✔
SQuAD 2.0 (Rajpurkar et al., 2018)	✗	✔	✗	✔	✔	✗
MS Marco (Nguyen et al., 2016)	✗	✔	✗	✗	✔	✔
NewsQA (Trischler et al., 2016)	✗	✔	✗	✔	✔	✔

[Choi et al., 2018]

Further Comments

Two ways to evaluate your project by yourself

- Practical values
  - How it helps me get some experience of using NLP tools?
  - How it helps me get some experience of training DL models?
  - Is it worth to put it on my resume when I am looking for a job/internship?

Further Comments

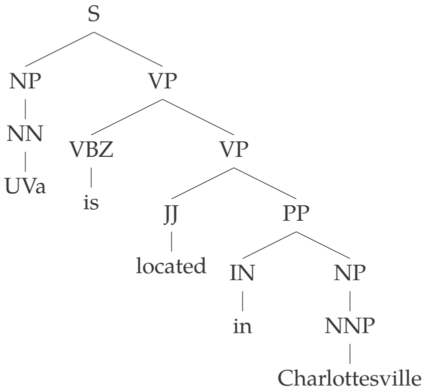
Two ways to evaluate your project by yourself

- Practical values
  - How it helps me get some experience of using NLP tools?
  - How it helps me get some experience of training DL models?
  - Is it worth to put it on my resume when I am looking for a job/internship?
- Research values
  - How this project is relevant to my research?
  - How this project help me understand resolving some real problems using NLP techniques?
  - Will this eventually lead to an academic paper?

Introduction

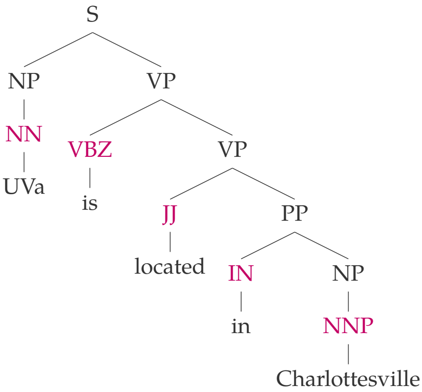
Parsing

- Input: `UVa is located in Charlottesville`
- Output:



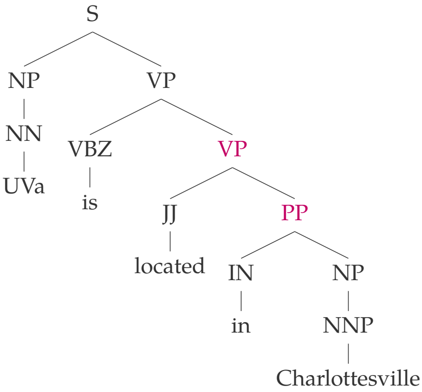
Information Conveyed by Parse Trees

Part of speech for each word



Information Conveyed by Parse Trees (II)

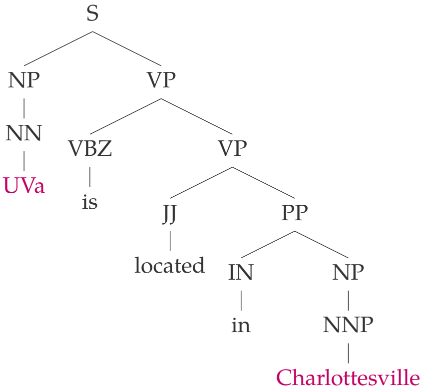
Phrase



- PP: *in Charlottesville*
- VP: *located in Charlottesville*

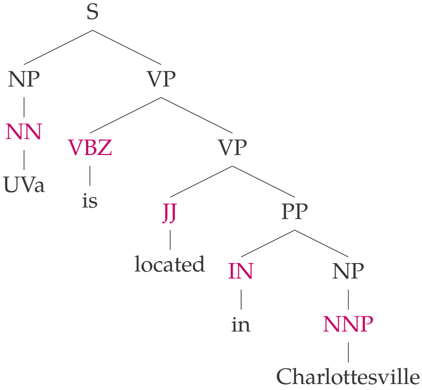
Information Conveyed by Parse Trees (III)

Useful relationships



Relationship between UVa and Charlottesville

An Example Application



- Question answering: *what is the location of UVa?*

Context-Free Grammars

Formal Definition

A context free grammar $G = (N, \Sigma, R, S)$ where

- N: a set of non-terminal symbols
- $S \in N$: a distinguished start symbol
- Σ: a set of terminal symbols
- R: a set of rules of the form $X \rightarrow Y_1 Y_2 \cdots Y_n$ for $n \geq 0$, $X \in N$, $Y_i \in N \cup \Sigma$

A Context-Free Grammar for English

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

S	$\rightarrow$	NP	VP
VP	$\rightarrow$	Vi	
VP	$\rightarrow$	Vt	NP
VP	$\rightarrow$	VP	PP
NP	$\rightarrow$	DT	NN
NP	$\rightarrow$	NP	PP
PP	$\rightarrow$	IN	NP

Vi	$\rightarrow$	sleeps
Vt	$\rightarrow$	saw
NN	$\rightarrow$	man
NN	$\rightarrow$	woman
NN	$\rightarrow$	telescope
DT	$\rightarrow$	the
IN	$\rightarrow$	with
IN	$\rightarrow$	in

(Left-Most) Derivations

S	→	NP	VP
VP	→	Vi	
VP	→	Vt	NP
VP	→	VP	PP
NP	→	DT	NN
NP	→	NP	PP
PP	→	IN	NP

Vi	→	sleeps
Vt	→	saw
NN	→	man
NN	→	woman
NN	→	telescope
DT	→	the
IN	→	with
IN	→	in

- Left-most derivation: always pick the left-most non-terminal symbol for replacement

An Example

S	$\rightarrow$	NP	VP
VP	$\rightarrow$	Vi	
VP	$\rightarrow$	Vt	NP
VP	$\rightarrow$	VP	PP
NP	$\rightarrow$	DT	NN
NP	$\rightarrow$	NP	PP
PP	$\rightarrow$	IN	NP

Vi	$\rightarrow$	sleeps
Vt	$\rightarrow$	saw
NN	$\rightarrow$	man
NN	$\rightarrow$	woman
NN	$\rightarrow$	telescope
DT	$\rightarrow$	the
IN	$\rightarrow$	with
IN	$\rightarrow$	in

Derivation	Rules used
S	S $\rightarrow$ NP VP
NP VP	

An Example

S	$\rightarrow$	NP	VP
VP	$\rightarrow$	Vi	
VP	$\rightarrow$	Vt	NP
VP	$\rightarrow$	VP	PP
NP	$\rightarrow$	DT	NN
NP	$\rightarrow$	NP	PP
PP	$\rightarrow$	IN	NP

Vi	$\rightarrow$	sleeps
Vt	$\rightarrow$	saw
NN	$\rightarrow$	man
NN	$\rightarrow$	woman
NN	$\rightarrow$	telescope
DT	$\rightarrow$	the
IN	$\rightarrow$	with
IN	$\rightarrow$	in

Derivation	Rules used
S	S $\rightarrow$ NP VP
NP VP	NP $\rightarrow$ DT NN
DT NN VP	

An Example

S	$\rightarrow$	NP	VP
VP	$\rightarrow$	Vi	
VP	$\rightarrow$	Vt	NP
VP	$\rightarrow$	VP	PP
NP	$\rightarrow$	DT	NN
NP	$\rightarrow$	NP	PP
PP	$\rightarrow$	IN	NP

Vi	$\rightarrow$	sleeps
Vt	$\rightarrow$	saw
NN	$\rightarrow$	man
NN	$\rightarrow$	woman
NN	$\rightarrow$	telescope
DT	$\rightarrow$	the
IN	$\rightarrow$	with
IN	$\rightarrow$	in

Derivation	Rules used
S	S $\rightarrow$ NP VP
NP VP	NP $\rightarrow$ DT NN
DT NN VP	DT $\rightarrow$ the
the NN VP	

An Example

S	$\rightarrow$	NP	VP
VP	$\rightarrow$	Vi	
VP	$\rightarrow$	Vt	NP
VP	$\rightarrow$	VP	PP
NP	$\rightarrow$	DT	NN
NP	$\rightarrow$	NP	PP
PP	$\rightarrow$	IN	NP

Vi	$\rightarrow$	sleeps
Vt	$\rightarrow$	saw
NN	$\rightarrow$	man
NN	$\rightarrow$	woman
NN	$\rightarrow$	telescope
DT	$\rightarrow$	the
IN	$\rightarrow$	with
IN	$\rightarrow$	in

Derivation	Rules used
S	S $\rightarrow$ NP VP
NP VP	NP $\rightarrow$ DT NN
DT NN VP	DT $\rightarrow$ the
the NN VP	NN $\rightarrow$ man
the man VP	

An Example

S	→	NP	VP
VP	→	Vi	
VP	→	Vt	NP
VP	→	VP	PP
NP	→	DT	NN
NP	→	NP	PP
PP	→	IN	NP

Vi	→	sleeps
Vt	→	saw
NN	→	man
NN	→	woman
NN	→	telescope
DT	→	the
IN	→	with
IN	→	in

Derivation	Rules used
S	S → NP VP
NP VP	NP → DT NN
DT NN VP	DT → the
the NN VP	NN → man
the man VP	VP → Vi
the man Vi	

An Example

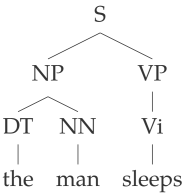
S	$\rightarrow$	NP	VP
VP	$\rightarrow$	Vi	
VP	$\rightarrow$	Vt	NP
VP	$\rightarrow$	VP	PP
NP	$\rightarrow$	DT	NN
NP	$\rightarrow$	NP	PP
PP	$\rightarrow$	IN	NP

Vi	$\rightarrow$	sleeps
Vt	$\rightarrow$	saw
NN	$\rightarrow$	man
NN	$\rightarrow$	woman
NN	$\rightarrow$	telescope
DT	$\rightarrow$	the
IN	$\rightarrow$	with
IN	$\rightarrow$	in

Derivation	Rules used
S	S $\rightarrow$ NP VP
NP VP	NP $\rightarrow$ DT NN
DT NN VP	DT $\rightarrow$ the
the NN VP	NN $\rightarrow$ man
the man VP	VP $\rightarrow$ Vi
the man Vi	Vi $\rightarrow$ sleeps
the man sleeps	

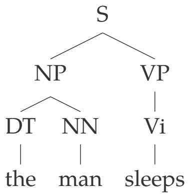
From Derivations to Parse Tree

Derivation	Rules used
S	S → NP VP
NP VP	
DT NN VP	DT → the
the NN VP	NN → man
the man VP	VP → Vi
the man Vi	Vi → sleeps
the man sleeps	



From Derivations to Parse Tree

Derivation	Rules used
S	S → NP VP
NP VP	NP → DT NN
DT NN VP	DT → the
the NN VP	NN → man
the man VP	VP → Vi
the man Vi	Vi → sleeps
the man sleeps	



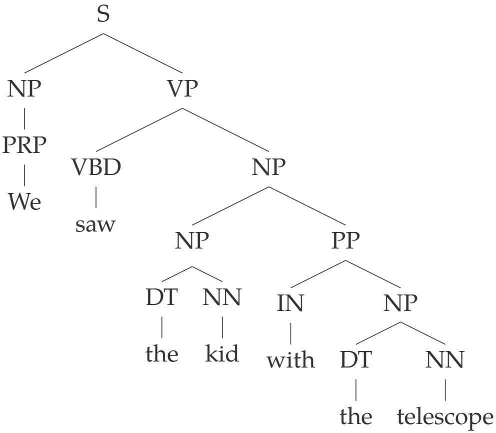
Properties of CFGs

- A CFG defines a set of possible derivations
- A string $s \in \Sigma^*$ is in the language defined by the CFG if there is at least one derivation that yield s
- Each string in the language generated by the CFG may have more than one derivation (“ambiguity”)

Ambiguity

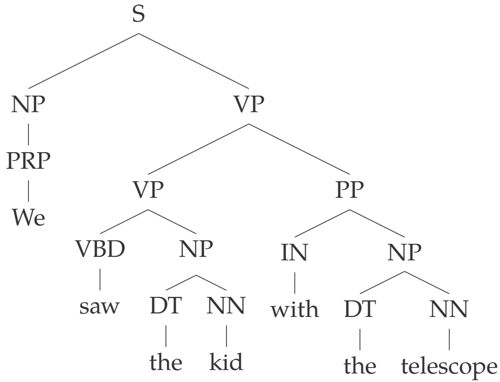
An Example of Ambiguity

Sentence: *We saw the kid with the telescope*



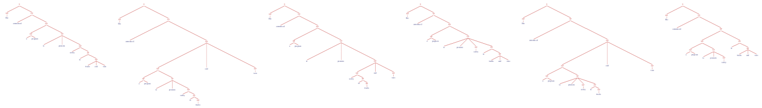
An Example of Ambiguity (II)

Sentence: *We saw the kid with the telescope*



Problem with Parsing: Ambiguity

Sentence: *She announced a program to promote safety in trucks and vans*



Sources of Ambiguity (I)

Part-of-Speech ambiguity

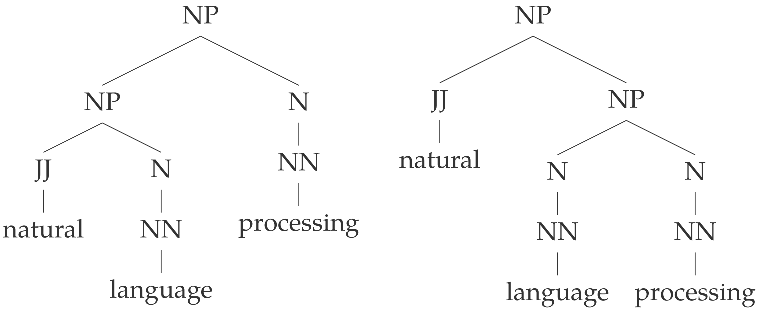
- NN $\longrightarrow$ duck
- Vi $\longrightarrow$ duck

[VP [Vt saw] [NP [PRP her] [NN duck]]]

[VP [Vt saw] [S [PRP her] [VP [Vi duck]]]]

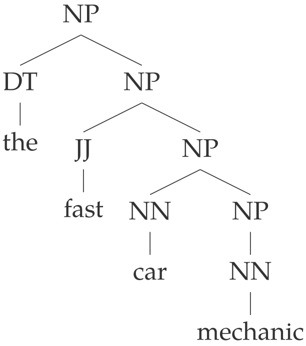
Sources of Ambiguity (II)

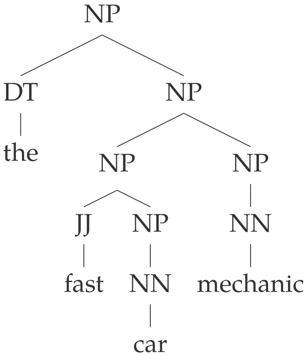
Noun premodifiers



Sources of Ambiguity (III)

Noun premodifiers





Probabilistic CFGs

A Probabilistic Context-Free Grammar (PCFG)

- N: a set of non-terminal symbols
- $S \in N$: a distinguished start symbol
- Σ: a set of terminal symbols

S	⇒	NP	VP	1.0
VP	⇒	Vi		0.4
VP	⇒	Vt	NP	0.4
VP	⇒	VP	PP	0.2
NP	⇒	DT	NN	0.3
NP	⇒	NP	PP	0.7
PP	⇒	P	NP	1.0

Vi	⇒	sleeps	1.0
Vt	⇒	saw	1.0
NN	⇒	man	0.7
NN	⇒	woman	0.2
NN	⇒	telescope	0.1
DT	⇒	the	1.0
IN	⇒	with	0.5
IN	⇒	in	0.5

Probability of a Tree

The probability of a tree $\boldsymbol{t}$ with rules

$$\alpha_1 \rightarrow \beta_1, \alpha_2 \rightarrow \beta_2, \ldots, \alpha_n \rightarrow \beta_n$$

is

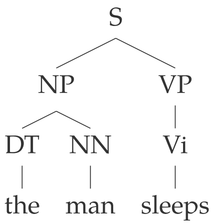
$$P(\boldsymbol{t}) = \prod_{i=1}^{n} P(\alpha_i \rightarrow \beta_i) \tag{1}$$

where $P(\alpha \rightarrow \beta)$ is the rule of $\alpha \rightarrow \beta$

An Example

S	$\Rightarrow$	NP	VP	1.0
VP	$\Rightarrow$	Vi		0.4
VP	$\Rightarrow$	Vt	NP	0.4
VP	$\Rightarrow$	VP	PP	0.2
NP	$\Rightarrow$	DT	NN	0.3
NP	$\Rightarrow$	NP	PP	0.7
PP	$\Rightarrow$	P	NP	1.0

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



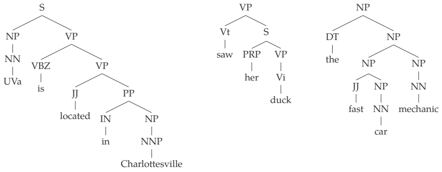
Properties of PCFGs

- Assigns a probability to each derivation, or parse-tree, allowed by the underlying CFG
- If one sentence has more than one derivations, we can rank them based on their probabilities
- The most likely parse tree for a sentence is

$$\arg\max_{\boldsymbol{t} \in \mathcal{T}(s)} P(\boldsymbol{t}) \tag{2}$$

Deriving a PCFG from a Corpus

- Given a set of example trees (a treebank), the underlying CFG can simply be all rules seen in the corpus



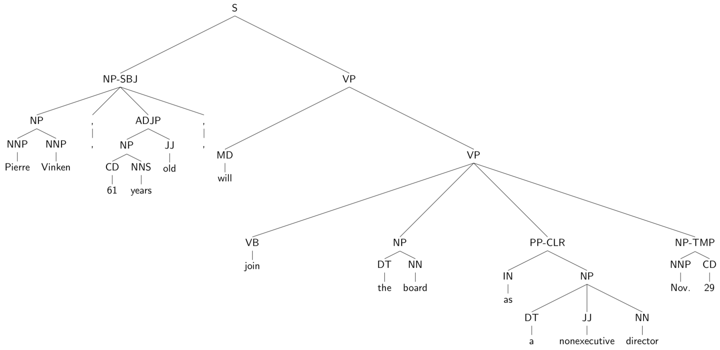
- Maximum likelihood estimates:

$$P(\alpha \rightarrow \beta) \approx \frac{c(\alpha \rightarrow \beta)}{c(\alpha)} \tag{3}$$

Penn Treebank

Penn Treebank

- 50,000 sentences with associated trees
- Usual setup: 40,000 training sentences, 2,400 test sentences

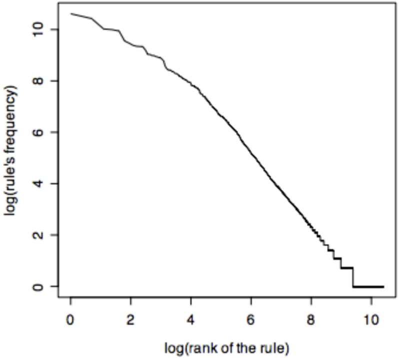


Some Penn Treebank Rules with Counts

40717 PP → IN NP
33803 S → NP-SBJ VP
22513 NP-SBJ → -NONE-
21877 NP → NP PP
20740 NP → DT NN
14153 S → NP-SBJ VP .
12922 VP → TO VP
11881 PP-LOC → IN NP
11467 NP-SBJ → PRP
11378 NP → -NONE-
11291 NP → NN
. . .
989 VP → VBG S
985 NP-SBJ → NN
983 PP-MNR → IN NP

100 VP → VBD PP-PRD
100 PRN → : NP :
100 NP → DT JJS
100 NP-CLR → NN
99 NP-SBJ-1 → DT NNP
98 VP → VBN NP PP-DIR
98 VP → VBD PP-TMP
98 PP-TMP → VBG NP
97 VP → VBD ADVP-TMP VP
. . .
10 WHNP-1 → WRB JJ
10 VP → VP CC VP PP-TMP
10 VP → VP CC VP ADVP-MNR
10 VP → VBZ S , SBAR-ADV
10 VP → VBZ S ADVP-TMP

Penn Treebank Rules: Statistics



Summary

1. About Group Project Proposal
2. Introduction
3. Context-Free Grammars
4. Ambiguity
5. Probabilistic CFGs
6. Penn Treebank

Reference

Choi, E., He, H., Iyyer, M., Yatskar, M., Yih, W.-t., Choi, Y., Liang, P., and Zettlemoyer, L. (2018).
Quac: Question answering in context.
arXiv preprint arXiv:1808.07036.

Collins, M. (2017).
Natural language processing: Lecture notes.

Smith, N. A. (2018).
Natural language processing: Lecture notes.