CSCE 771: Computer Processing of Natural Language
Lecture 7: Statistical Parsing, Quiz

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

8TH SEPTEMBER, 2022

Carolinian Creed: "I will practice personal and academic integrity."

Acknowledgement: Used materials by Jurafsky & Martin,

Organization of Lecture 7

- Opening Segment
  - Recap of Last Class
  - Announcements
- Main Lecture

Main Section
- Statistical Parsing
- Quiz 1

- Concluding Segment
  - About Next Lecture – Lecture 8

Recap of Lecture 6

- We reviewed projects
  - Those without ‘T’ must choose a topic and get reviewed by next class
- We discussed parsers
  - Shallow parsers
  - Dependency parsers

6	Sep 8 (Th)	Statistical Parsing, QUIZ
7	Sep 13 (Tu)	Review Parsing, Quiz review, Review Project, Introduce Evaluation
8	Sep 15 (Th)	Language Model – Vector embeddings, CNN/ RNN
9	Sep 20 (Tu)	Semantics
10	Sep 22 (Th)	Review: Machine Learning for NLP, Evaluation - Metrics
11	Sep 27 (Tu)	Guest Lecture – Dr. Amitava Das: Glove, Word2Vec, Transformer Review: Reasoning for NLP
12	Sep 29 (Th)	Representation: Ontology, Knowledge Graph, QUIZ
13	Oct 4 (Tu)	Representation: Embeddings, Language Models
14	Oct 6 (Th)	Entity extraction
15	Oct 11 (Tu)	Guest Lecture – Dr. Amitava Das: Using lang models to solve NLP tasks

Announcements

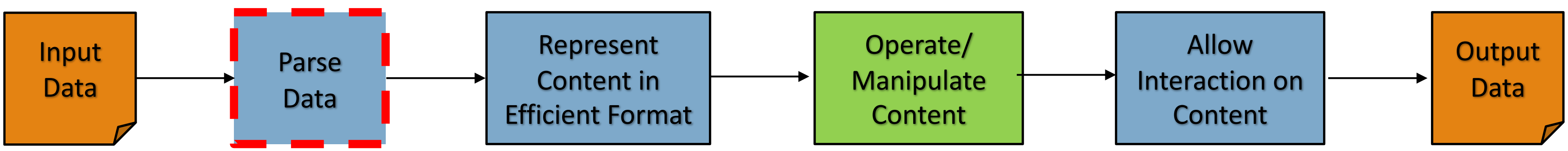
GUEST LECTURES ON LANGUAGE MODELS BY DR. AMITAVA DAS

Main Lecture

Statistical Parsing

Given a sentence X, predict the most **probable** parse tree Y

$$\underset{Y}{\operatorname{argmax}}\ P\ (Y|X)$$



Probabilistic CFG

N	a set of **non-terminal symbols** (or **variables**)
Σ	a set of **terminal symbols** (disjoint from N)
R	a set of **rules** or productions, each of the form $A \rightarrow \beta\ [p]$, where A is a non-terminal, β is a string of symbols from the infinite set of strings $(\Sigma \cup N)*$, and p is a number between 0 and 1 expressing $P(\beta\|A)$
S	a designated **start symbol**

p is the probability that non-terminal **A** will be expanded to the sequence **β**

$$\sum_{\beta} P(A \rightarrow \beta) = 1$$

A PCFG is said to be **consistent** if the sum of the probabilities of all sentences in the language equals 1

From Jurafsky & Martin

Probabilistic CFG Example

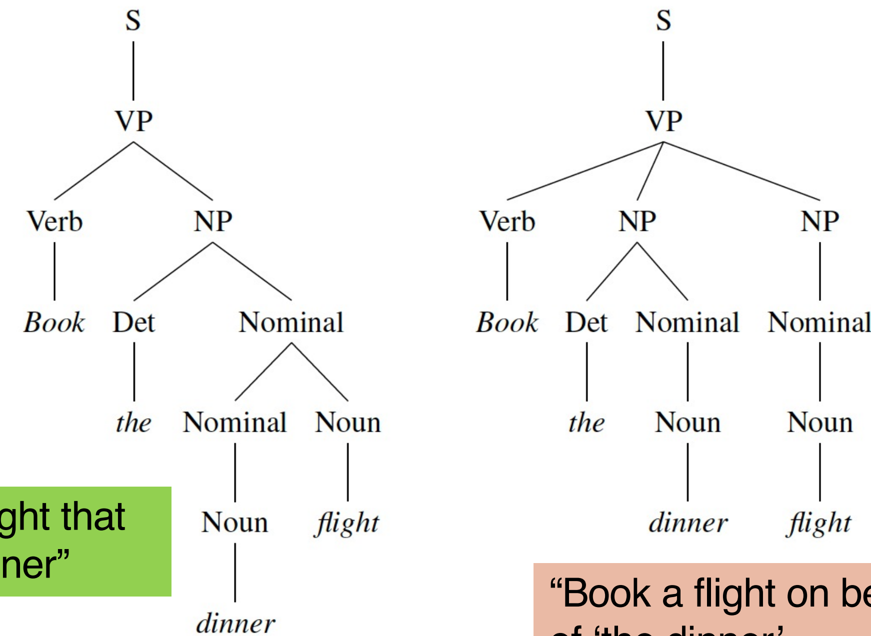
Grammar		Lexicon
$S \rightarrow NP\ VP$	[.80]	$Det \rightarrow that$ [.10] | a [.30] | the [.60]
$S \rightarrow Aux\ NP\ VP$	[.15]	$Noun \rightarrow book$ [.10] | $flight$ [.30]
$S \rightarrow VP$	[.05]	| $meal$ [.05] | $money$ [.05]
$NP \rightarrow Pronoun$	[.35]	| $flight$ [.40] | $dinner$ [.10]
$NP \rightarrow Proper\text{-}Noun$	[.30]	$Verb \rightarrow book$ [.30] | $include$ [.30]
$NP \rightarrow Det\ Nominal$	[.20]	| $prefer$ [.40]
$NP \rightarrow Nominal$	[.15]	$Pronoun \rightarrow I$ [.40] | she [.05]
$Nominal \rightarrow Noun$	[.75]	| me [.15] | you [.40]
$Nominal \rightarrow Nominal\ Noun$	[.20]	$Proper\text{-}Noun \rightarrow Houston$ [.60]
$Nominal \rightarrow Nominal\ PP$	[.05]	| NWA [.40]
$VP \rightarrow Verb$	[.35]	$Aux \rightarrow does$ [.60] | can [.40]
$VP \rightarrow Verb\ NP$	[.20]	$Preposition \rightarrow from$ [.30] | to [.30]
$VP \rightarrow Verb\ NP\ PP$	[.10]	| on [.20] | $near$ [.15]
$VP \rightarrow Verb\ PP$	[.15]	| $through$ [.05]
$VP \rightarrow Verb\ NP\ NP$	[.05]	
$VP \rightarrow VP\ PP$	[.15]	
$PP \rightarrow Preposition\ NP$	[1.0]	

Question: is the *PCFG in example consistent?*

From Jurafsky & Martin

Example

Interpretations of
"***Book the dinner flight***"



"Book a flight that serves dinner"

"Book a flight on behalf of 'the dinner'"

Rules			P	Rules			P
S	→	VP	.05	S	→	VP	.05
VP	→	Verb NP	.20	VP	→	Verb NP NP	.10
NP	→	Det Nominal	.20	NP	→	Det Nominal	.20
Nominal	→	Nominal Noun	.20	NP	→	Nominal	.15
Nominal	→	Noun	.75	Nominal	→	Noun	.75
				Nominal	→	Noun	.75
Verb	→	book	.30	Verb	→	book	.30
Det	→	the	.60	Det	→	the	.60
Noun	→	dinner	.10	Noun	→	dinner	.10
Noun	→	flight	.40	Noun	→	flight	.40

From Jurafsky & Martin

Decisions with PCFG

Probability of parse tree T, given sentence S, is

$$P(T,S) = \prod_{i=1}^{n} P(RHS_i|LHS_i)$$

Definition:
Yield of a parse tree = String of words allowed by parse tree

Of all parse trees with a yield of S, the disambiguation algorithm for parsing picks the parse tree that is most probable given S:

$$\hat{T}(S) = \underset{T s.t. S=\text{yield}(T)}{\text{argmax}} P(T|S) \longrightarrow \hat{T}(S) = \underset{T s.t. S=\text{yield}(T)}{\text{argmax}} P(T)$$

From Jurafsky & Martin

choosing the parse with the highest probability

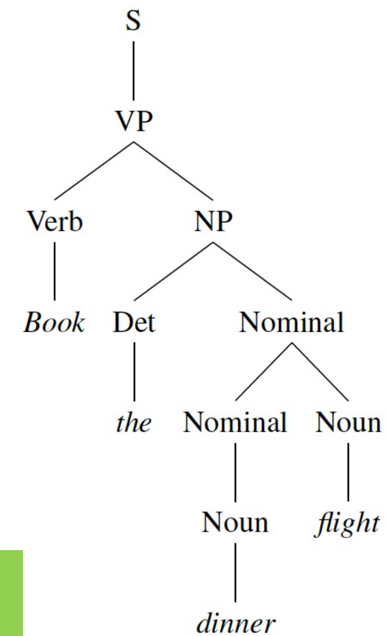
Interpretations of
"***Book the dinner flight***"

"Book a flight that serves dinner"

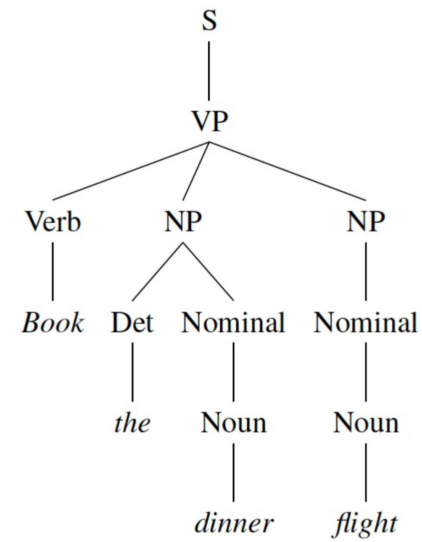
"Book a flight on behalf of 'the dinner'"

Example

Interpretations of
"***Book the dinner flight***"



"Book a flight that serves dinner"



"Book a flight on behalf of 'the dinner'"

$$P(T_{left}) = .05 * .20 * .20 * .20 * .75 * .30 * .60 * .10 * .40 = \mathbf{2.2 \times 10^{-6}}$$ ✓

$$P(T_{right}) = .05 * .10 * .20 * .15 * .75 * .75 * .30 * .60 * .10 * .40 = \mathbf{6.1 \times 10^{-7}}$$

Rules			P
S	→	VP	.05
VP	→	Verb NP	.20
NP	→	Det Nominal	.20
Nominal	→	Nominal Noun	.20
Nominal	→	Noun	.75
Verb	→	book	.30
Det	→	the	.60
Noun	→	dinner	.10
Noun	→	flight	.40

Rules			P
S	→	VP	.05
VP	→	Verb NP NP	.10
NP	→	Det Nominal	.20
NP	→	Nominal	.15
Nominal	→	Noun	.75
Nominal	→	Noun	.75
Verb	→	book	.30
Det	→	the	.60
Noun	→	dinner	.10
Noun	→	flight	.40

From Jurafsky & Martin

Assumptions/ Issues with PCFG - 1

Issue: CFG rules impose an independence assumption on probabilities that miss rule dependencies

- Example:
  - **nouns** can be **subjects** as well as **objects**
  - A **pronoun** is a **noun**, but also is a **determiner noun.** **[Example: NP -> DT NN :28, NP -> PRP 0.25]**
  - **Subjects** are **more likely** to be **pronouns** than **objects**. **[**91% subjects are pronouns, 34% objects are pronouns in Switchboard dataset]
- Same rule's application can be contextual based on where the rule is being applied. Example, NP -> PRP
- Not being able to differentiate can cause incorrect parsing

From Jurafsky & Martin

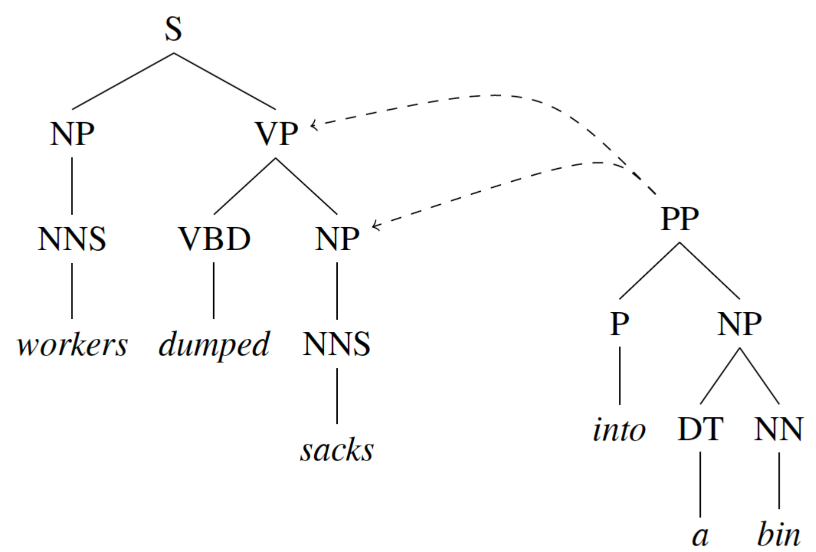
Assumptions/ Issues with PCFG - 2

Issue: Lack of sensitivity to lexical dependencies

Example: *worker dumped sacks into a bin*

"into a bin" prepositional phrase can be attached to either the VP or NP leading to different meanings

- When attached to VP, sacks are in location "into a bin"
- When attached to NP, "sacks into a bin" are dumped - nonsensical



From Jurafsky & Martin

Improvement: Probabilistic Lexicalized CFGs

- Augment PCFG with a lexical head for each rule.
- The probability of a rule is conditional on the lexical head

VP -> VBD NP P *is modified to*

VP(dumped,VBD) -> VBD(dumped,VBD) NP(sacks,NNS) PP(into,P)

From Jurafsky & Martin

Calculating Probability from Treebank

$$P(\alpha \to \beta|\alpha) = \frac{\text{Count}(\alpha \to \beta)}{\sum_{\gamma}\text{Count}(\alpha \to \gamma)} = \frac{\text{Count}(\alpha \to \beta)}{\text{Count}(\alpha)}$$

Probability of each expansion of a non-terminal:

- counting the number of times an expansion occurs
- normalizing for all expansions

From Jurafsky & Martin

Evaluating Parsers - PARSEVAL

Degree to which the constituents in the hypothesis parse tree look like the constituents in a hand-labeled, gold-reference parse like PENN TreeBank

Overall measure is by F1 score

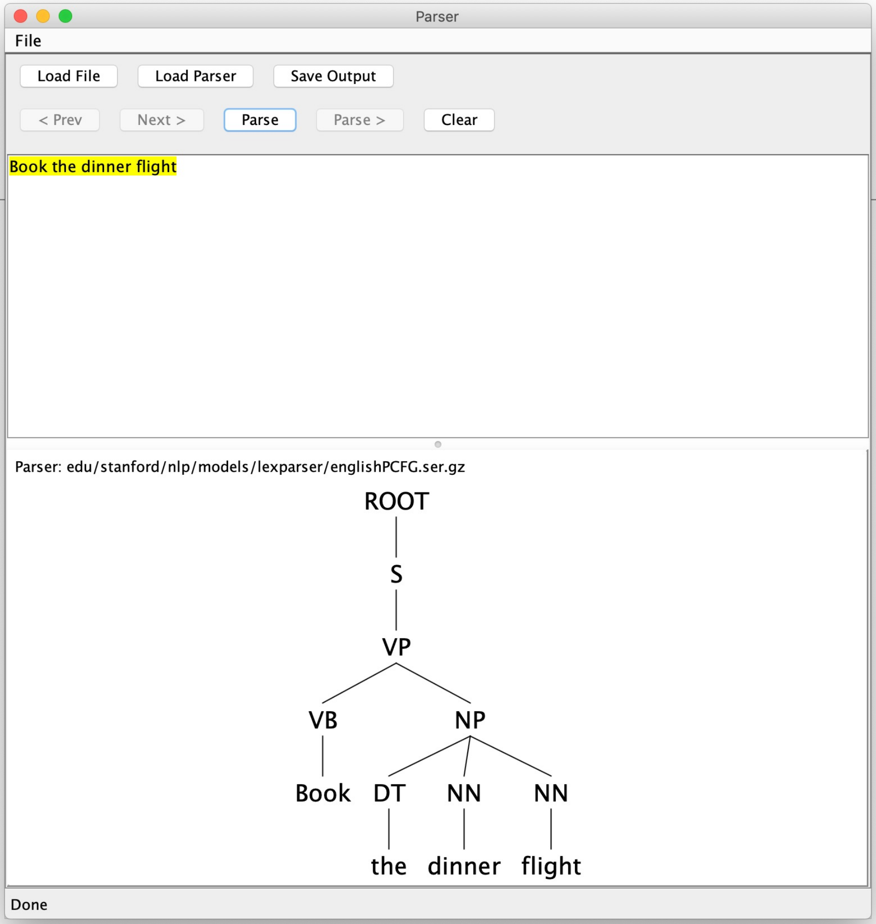
$$F_1 = \frac{2PR}{P+R}$$

$$\textbf{labeled recall:} = \frac{\text{\# of correct constituents in hypothesis parse of } s}{\text{\# of correct constituents in reference parse of } s}$$

$$\textbf{labeled precision:} = \frac{\text{\# of correct constituents in hypothesis parse of } s}{\text{\# of total constituents in hypothesis parse of } s}$$

From Jurafsky & Martin

Output from a Popular Parser: Stanford Parser

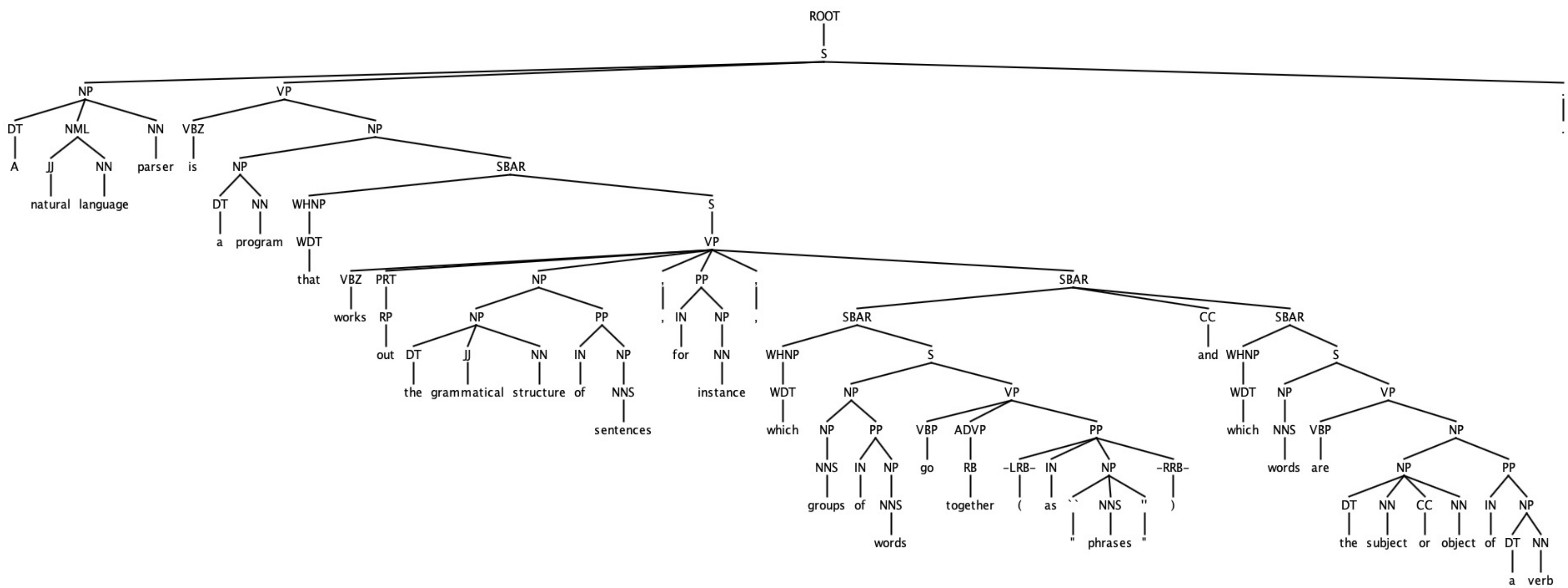


Demonstrations in multiple languages

https://nlp.stanford.edu/software/lex-parser.shtml

Stanford Parser Example - 2

A natural language parser is a program that works out the grammatical structure of sentences, for instance, which grou of words go together (as "phrases") and which words are th subject or object of a verb.



Lecture 7: Concluding Comments

- We have completed parsing
- Probabilistic grammars
  - assign a probability to a sentence or string of words
  - In a probabilistic context-free grammar (PCFG), every rule is annotated with the probability of that rule being chosen assuming conditional independence.
  - The probability of a sentence is computed by multiplying the probabilities of each rule in the parse of the sentence.
- Probabilistic lexicalized CFGs:
  - PCFG model is augmented with a lexical head for each rule.

Concluding Segment

QUIZ

- In class

About Next Lecture – Lecture 8

Lecture 8: Reviewing Recent Topics

- Review quiz
- Review parsing
- Review projects
- Introduce evaluation metrics in NLP context