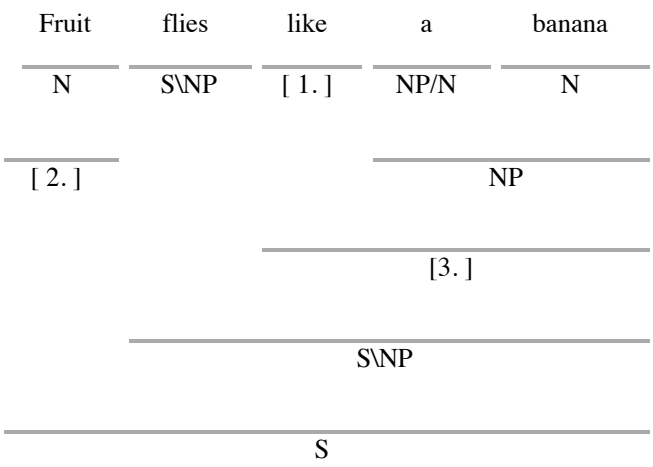
Question 1:

Consider the sentence *Fruit flies like a banana*.

a. This sentence is ambiguous. Paraphrase the two possible interpretations

b. Consider the following CCG parses of the sentence, which have some of their labels removed (indicated with [ ]). For each parse, fill in the missing labels.

Parse 1:



1.
2.
3.

Parse 2:

Fruit	flies	like	a	banana
N	N\N	[4.]	NP/N	N
N			NP	
[5.]		[6.]		
S				

4.

5.

6.

c. For each parse, indicate which of the two meanings the given parse corresponds to.

Question 2:

Consider the following HMM model for tagging. We will assume that the vocabulary consists of three words, *the, dog, sleeps*. There are two possible part-of-speech tags, *D, N*.

One set of parameters in the HMM are probabilities of the form P(*word*|*tag*), for example P(*the*| *D*), P(*the*| *N*), etc. In our HMM these probabilities take the following values:

Parameter	Value
$P(the\|D)$	0.7
$P(dog\|D)$	0.2
$P(sleeps\|D)$	0.1
$P(the\|N)$	0.2
$P(dog\|N)$	0.5
$P(sleeps\|N)$	0.3

Another set of parameters in the HMM are of the form P(*tag*|*previous-tag*), for example P(*N*|*D*), P(*D*|*N*), etc. Note that we have a bigram HMM tagger, in that each tag depends only on the previous tag. We take START to be a special tag which always appears at the start of a sentence, and STOP to be a tag that appears at the end of the sentence. In our model these parameters take the following values:

Parameter	Value
$P(D\|START)$	0.1
$P(N\|START)$	0.2
$P(STOP\|START)$	0.7
$P(D\|D)$	0.6
$P(N\|D)$	0.2
$P(STOP\|D)$	0.2
$P(D\|N)$	0.4
$P(N\|N)$	0.5
$P(STOP\|N)$	0.1

We would like you to define a PCFG that is "equivalent" to the above HMM. By "equivalent" we mean the following:
• For any symbol sequence x and state sequence y which has probability $P_{HMM}(x, y) > 0$ under the HMM: 1) there should be a parse tree y' such that the pair x, y' gets probability $P_{PCFG}(x, y')$ under the PCFG; 2) this probability should satisfy $P_{HMM}(x, y) = P_{PCFG}(x, y')$
• There should be a one-to-one function f(y) = y' that maps a state sequence in the HMM to a parse tree generated by the PCFG, and that satisfies $P_{HMM}(x, y) = P_{PCFG}(x, f(y))$

In your solution you should: (a) write down your PCFG which is equivalent to the above HMM; (b) define the function f(y) between state sequences and parse trees.
Note: you may find it useful to make use of productions in your PCFG where ϵ (the empty string) is on the right-hand-side of your rule. For example,

$$X \rightarrow \epsilon with P(X \rightarrow \epsilon | X) = 0.1$$

states that the non-terminal X can rewrite to the empty string with probability 0.1. To elaborate further, the context-free grammar

$$S \rightarrow aX$$
$$X \rightarrow \epsilon$$

Generates one string, i.e., the string a.

Question 3:
Consider the following context-free grammar that recognizes simple sentences such as "the students teach the students":

Grammar:
S --> NP VP
NP --> DET N
NP --> N
VP --> V
VP --> V NP

Lexicon:
DET --> the
N --> student
N --> students
V --> look
V --> looks
V --> teach
V --> teaches

Part A: While this grammar parses all grammatical sentences for the given lexicon, it also parses many ungrammatical sentences. For example, sentences with subject/verb disagreement, such as "the student teach", are parsed. In addition to sentences with subject/verb disagreement, identify two types of ungrammatical sentences that can be parsed with the given grammar and provide an example for each.

Part B: We would like to augment the basic grammar with unification constraints such that all grammatical sentences unify, producing valid parses, while ungrammatical sentences fail unification. In a paragraph, describe how the grammar could be augmented with unification constraints to address subject/verb disagreement and the additional types of ungrammatical sentences you identified above. Clearly indicate what features should be added to the lexical entries and what constraints should be added to the grammar rules.

Question 4:

We covered language models in class. Please recall the n-gram models that were used in Homework 1.

(a) We toss a fair coin a million times. Please estimate the unigram, bigram, and trigram probabilities for each of H and T (heads and tails), e.g., H, HT, TTH, etc.

(b) Based on your estimation in (a), which n-gram model do you think works better in estimating the next outcome after the following outcomes: **T H H T H T H T T**?
A. unigram is better
B. bigram is better
C. trigram is better
D. None of the above

Is the answer the same or different compared to what you found in homework 1? Please explain why.

(c) In Homework 1, how did you evaluate how good the model is (e.g., by what method or algorithm)?
A. MLE
B. Entropy
C. Word Error Rate
D. Perplexity
E. Precision and Recall

(d) True or False: If in Homework 1 you were instructed to work with 50-grams, the result would be significantly better than 1-, 2-, 3-grams.

(e) To work with unseen data, we use smoothing. Using add-one (Laplace) smoothing, please calculate the smoothed unigram of "Pikachu" probability in the following corpus (It's ok to leave it in fractions rather than in decimals):

Snorlax Charmander Bulbasaur Pikachu Pikachu Squirtle Snorlax Snorlax Charmander Pikachu

(f) Please choose the technique that can be used for dealing with sparse data.
A. Backoff
B. Interpolation
C. Regularization
D. Kneser-Ney
E. A and B

(g) True or False (You will get extra marks if you get this right, but if you get this wrong, you will get marks deducted from your score. Leaving it blank will not affect your score).

$$\sum_{n} C(w_{n-1}w_n) = C(w_{n-1})$$

Where C is the count of certain words in corpus, and w_n is the word appearing after w_{n-1}.

SOLUTIONS:

Solution 1:

Consider the sentence *Fruit flies like a banana*.

a. This sentence is ambiguous. Paraphrase the two possible interpretations.

1. Fruit flies (insect) enjoy/have a preference for a banana.
2. Fruit (food) flies (verb) in the way a banana flies.

b. Consider the following CCG parses of the sentence, which have some of their labels removed (indicated with [ ]). For each parse, fill in the missing labels.

Parse 1:

1. ((S\NP)\(S\NP))/NP
2. NP
3. (S\NP)\(S\NP)

Parse 2:

4. (S\NP)/NP
5. NP
6. S\NP

c. For each parse, indicate which of the two meanings the given parse corresponds to.
Parse 1 = meaning 2 in a

Parse 2 = meaning 1 in a

SOLUTION 2:

PCFG:

S → START D’	0.1
S → START N’	0.2
S → START STOP	0.7
D’ → D D’	0.6
D’ → D N’	0.2
D’ → D STOP	0.2
N’ → N D’	0.4
N’ → N N’	0.5
N’ → N STOP	0.1
D → the	0.7
D → dog	0.2
D → sleeps	0.1
N → the	0.2
N → dog	0.5
N → sleeps	0.3
START → *	1.0
STOP → *	1.0

f(y):

Let $s_1, s_2, \ldots, s_n$ be the state sequence where $s_i \in \{D, N\}$ and $w_1, w_2, \ldots, w_n$ be the corresponding symbol sequence. Then return the following tree:

SOLUTION 3:

Part A:

The following types of ungrammatical sentences parse:

- Type: sentences with incorrect definiteness / determiner usage
  - Example: “student teaches”
- Type: sentences with incorrect verb transitiveness
  - Example: “students look the student”

Part B:

In the lexicon, nouns can be augmented with number (num = sg/pl), and verbs can be augmented with usage (use = sg/pl) and transitiveness (trans = +/-). Rules building a NP from a single N should disallow singular nouns (N.num = pl). Rules combining a V and NP should require V to be transitive (V.trans = +). Rules combining a NP and VP should require NP.N’s number to agree with VP.V’s usage.

SOLUTION 4:

(a) Should be ½ for all of them.

(b) D. the n-grams are equally useless.
This is different from homework 1. Because this is totally random, while language is not.

(c) D. Perplexity

(d) False. Because data is sparse in that case, and Markov assumption.

(e) Assign probability mass to unseen data. (3 + 1) / (10 + 5) = 4 / 15

(f) E. Backoff or Interpolation

(g) True