CS 274A Natural Language Processing (Spring 2023), Final Exam

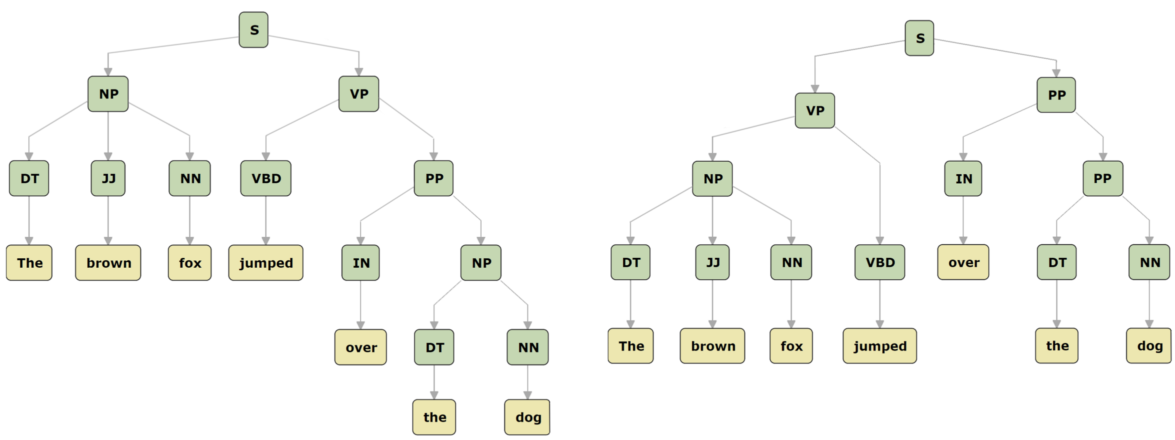
Instructions

- Time: 10:15–11:45am (90 minutes)
- This exam is closed-book, but you may bring one A4-size cheat sheet. Put all the study materials and electronic devices (with the exception of a calculator) into your bag and put your bag in the front, back, or sides of the classroom.
- You can write your answers in either English or Chinese.
- Two blank pieces of paper are attached, which you can use as scratch paper. Raise your hand if you need more paper.
- For multiple choice questions:
  - □ means you should mark ALL choices that apply;
  - ○ means you should mark exactly ONE choice;
  - When marking a choice, please fill in the bubble or square COMPLETELY (e.g., ● and ■). Ambiguous answer will receive no points.
  - For each question with □ choices, you get half of the points for selecting a non-empty proper subset of the correct answers.

1 Constituency parsing (20 pt)

1.1 Parser Evaluation (3 pt)

In constituency parsing, metrics for evaluating a parser include precision, recall, and F1. Please compute the precision, recall, and F1 of the following predicted parse tree against the ground truth parse tree:



ground truth parse tree   predicted parse tree

$P =$__________, $R =$__________, $F1 =$__________.

1.2 Span-based Constituency Parsing (3 pt)

Select all correct statement(s):

- ☐ CKY used in span-based constituency parsing is a bottom-up dynamic programming algorithm.
- ☐ In span-based constituency parsing, the predicted label of each span is the label with the highest score.
- ☐ Span-based constituency parser is a generative parser.
- ☐ With a grammar with N nonterminals and a sentence with L words, a span-based constituency parser needs to compute $NL(L-1)$ scores in total.

1.3 Rule-based Constituency Parsing (8 pt)

1.3.1 CYK with CFG (5 pt)

Suppose you are given the following grammar:

```
S    ->  N VP                    N    ->  him
N    ->  N PP                    N    ->  telescope
VP   ->  V N                     N    ->  I
VP   ->  VP PP                   V    ->  saw
PP   ->  P N                     P    ->  with
```

Please use CKY to parse the sentence, "I saw him with telescope" with the above grammar and complete the following table.

	1	2	3	4	5
0					
1					
2					
3					
4					

This sentence is ambiguous under this grammar. When parsing with CKY, the ambiguity exists when computing cell ______________ (in format "i, j", i.e., row index and then column index). If there are more than one ambiguous cells, just write one of them.

1.3.2 CYK with PCFG (3 pt)

To resolve the ambiguity in the grammar, we assign each grammar rule with a probability as follows:

```
S    ->  N VP     1.0            N    ->  him          0.2
N    ->  N PP     0.3            N    ->  telescope    0.2
VP   ->  V N      0.4            N    ->  I            0.3
VP   ->  VP PP    0.6            V    ->  saw          1.0
PP   ->  P N      1.0            P    ->  with         1.0
```

Then, the probability of the best parse of the sentence is _________, the probability of the second best parse is _________.

- ○ 0.00768, 0.00384
- ○ 0.00288, 0.00144
- ○ 0.00216, 0.00108
- ○ 0.00144, 0.00072

1.4 Inside-Outside Algorithm (3 pt)

Select all correct statement(s):

- ☐ The inside probability $\beta_j(p, q)$ is the probability that the sequence from position p to q is generated by nonterminal N_j.
- ☐ The inside-outside algorithm is used in unsupervised learning of the parameters of a PCFG.
- ☐ By replacing the *max* operation in CYK with *sum*, we get an algorithm that computes the outside probabilities.
- ☐ The outside probability $\alpha_j(1, |X|)$ of sentence X is 1 if N_j is the start symbol and 0 if N_j is not the start symbol. Here we use the [inclusive, inclusive] span index.

1.5 Transition-based Parsing (3 pt)

Select all correct statement(s) about bottom-up transition-based parsing taught in class:

- ☐ If a transition-based parser is trained only on correct data, it may have unexpected behavior in later transitions after it makes a mistake.
- ☐ The time complexity of a transition-based parser is linear to the sequence length.
- ☐ We can use beam search in transition-based parsing, which often improves the performance of the parser.
- ☐ A correct transition sequence must end with REDUCE-S, where S is the start symbol.

2 Dependency Parsing (20 pt)

In this question, we will analyze the dependency parse of the sentence: "I saw her duck".

2.1 Basics (8 pt)

Figure 1 shows two dependency parse trees.

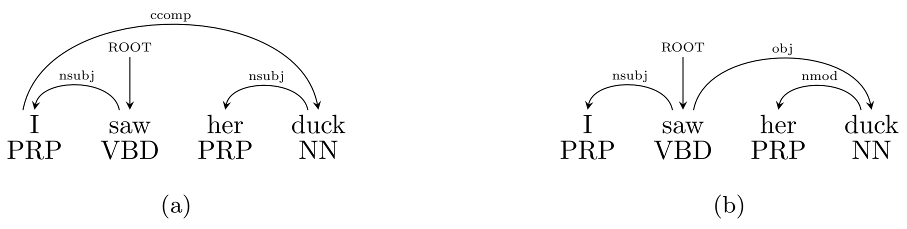


Figure 1: Two dependency parses of the sentence "I saw her duck".

2.1.1 Head and Dependent (2 pt)

Consider the edge with label 'nsubj' pointing from 'saw' to 'I'. Which one of the following statements is true?

- ○ 'I' is the head, 'saw' is the dependent.
- ○ 'I' is the dependent, 'saw' is the head.
- ○ Both 'I' and 'saw' are heads.
- ○ Both 'I' and 'saw' are dependents.

2.1.2 Projectivity (2 pt)

Are these dependency parse trees projective?

- ○ (a) is projective; (b) is not projective.
- ○ (a) is not projective; (b) is projective.
- ○ Both of them are projective.
- ○ Neither of them are projective.

2.1.3 Evaluation (2 pt)

Suppose (a) is the predicted parse tree, while (b) is the gold parse tree. What is the Unlabelled Attachment Score (UAS)? What about the Labelled Attachment Score (LAS)?

- ○ UAS = 75%, LAS = 50%
- ○ UAS = 50%, LAS = 75%
- ○ UAS = 66%, LAS = 33%
- ○ UAS = 33%, LAS = 66%

2.1.4 Dependency and Constituency (2pt)

Suppose we want to convert Figure 1(b) into a constituency parse tree. Figure 2 shows two constituency trees of the sentence "I saw her duck". Without differentiating left-branching and right-branching, which is a possible constituency parse tree produced by Figure 1(b)?

- ○ Only (a) is possible.
- ○ Only (b) is possible.
- ○ Both (a) and (b) are possible.

2.2 Graph-Based Dependency Parsing (8 pt)

Table 1 shows the score matrix for sentence "I saw her duck". Answer the following questions.

	I	saw	her	duck
ROOT	0	0	7	8
I	-	4	2	0
saw	2	-	8	6
her	10	5	-	2
duck	8	6	9	-

Table 1: A score matrix for sentence "I saw her duck".

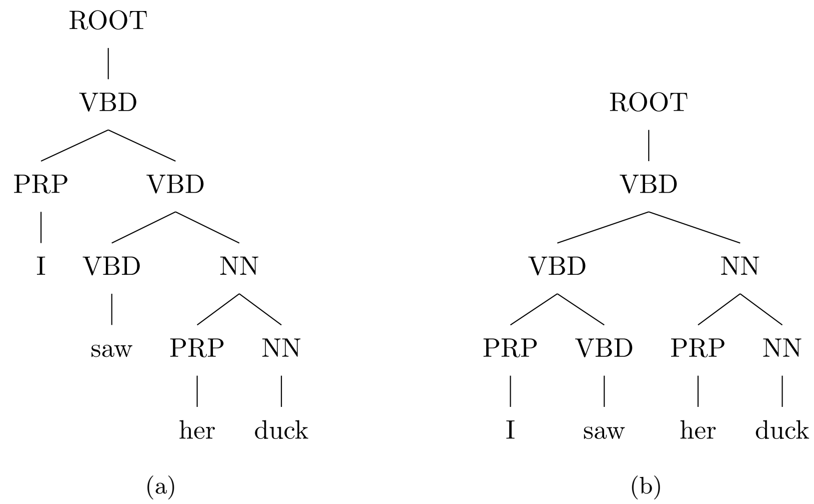


Figure 2: Two constituency parses of the sentence “I saw her duck”.

2.2.1 The Eisner Algorithm (1) (2pt)

Recall in Eisner, triangles represent a partial tree whose root is the word at the right angle and no words except root expect more children. What is the score of the left triangle containing words “saw her”, as shown in Figure 3?

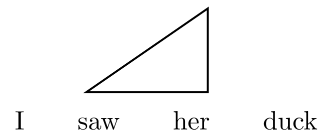


Figure 3: An illustration for eisner score.

○ 0 ○ 5 ○ 6 ○ 8 ○ 9 ○ 13

2.2.2 The Eisner Algorithm (2) (2pt)

Given the arc scores in Table 1, what’s the head of “saw” if we run the Eisner Algorithm?

○ I ○ saw ○ her ○ duck ○ ROOT

2.2.3 The Chu-Liu-Edmonds Algorithm (2pt)

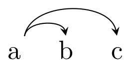
Recall in Chu-Liu-Edmonds, we contract the graph then expand it. Suppose we pick nodes from left to right, given the arc scores in Table 1, what is the number of times we need to contract the graph?

○ 0 ○ 1 ○ 2 ○ 3 ○ 4

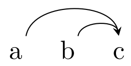
2.2.4 Second-order Graph-based Dependency Parsing (2pt)

In second-order graph-based dependency parsing, each connected pair of arcs has a score. The tree score is the sum of arc-pair scores. Below are three arc-pair patterns, which of them are considered in second-order graph-based dependency parsing?

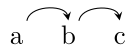
○ sibling, coparent



(a) sibling



(b) coparent



(c) grandparent

Figure 4: Three patterns for arc-pairs.

- ○ coparent, grandparent
- ○ sibling, grandparent
- ○ sibling, coparent, grandparent

2.3 Transition-Based Dependency Parsing (4 pt)

2.3.1 Transitions (2 pt)

Suppose Figure 5 shows the top two elements of the stack S during transition-based dependency parsing. Which element in Figure 6 will be the top element on the stack S if we apply action 'RIGHT-ARC'?

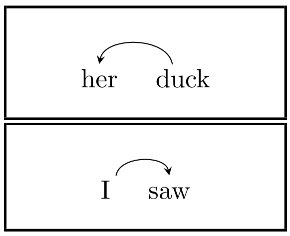
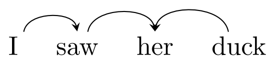
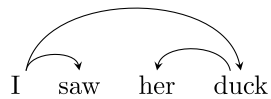


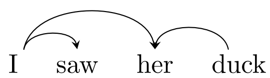
Figure 5: The top two elements on the stack S.



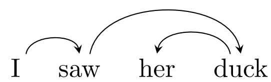
(a)



(b)



(c)



(d)

Figure 6: Three patterns for arc-pairs.

○ (a) ○ (b) ○ (c) ○ (d)

2.3.2 Number of Actions (2 pt)

How many actions ('SHIFT', 'LEFT-ARC', 'RIGHT-ARC') should we take to perform transition-based dependency parsing on sentence "I saw her duck"?

○ 0 ○ 1 ○ 4 ○ 7 ○ 8 ○ 16

3 Semantics (10 pt)

3.1 Lexical Semantics (5 pt)

3.1.1 Semantic Relations (3 pt)

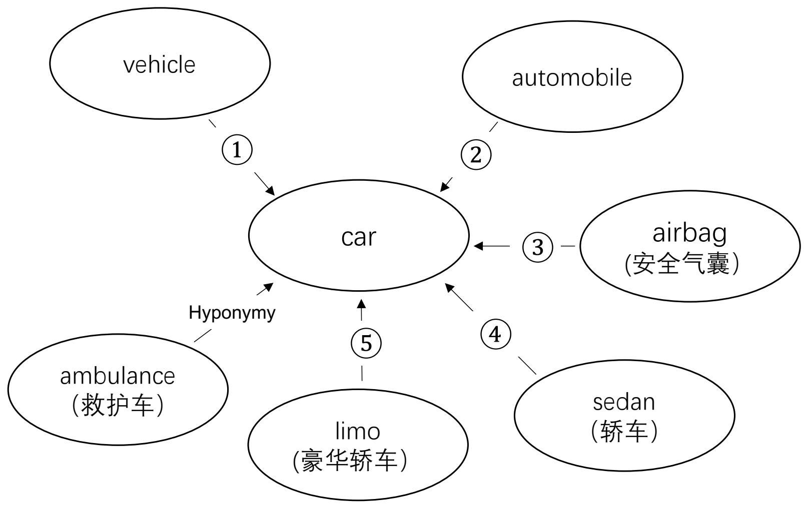


Figure 7: Semantic Relation Graph

Given a semantic relation graph in Fig. 7, please select all correct match(es). Reminder: Hyponymy (Subset, Is-a relation), Hypernymy (Superset), and Meronymy (Part-of relation):

□ ① Hypernymy.

□ ② Meronymy

□ ③ Meronymy

□ ④ Hyponymy

□ ⑤ Hypernymy

3.1.2 WordNet and Word Sense Disambiguation (2 pt)

Select all correct statement(s) about WordNet and Word Sense Disambiguation:

□ Words inside a synset are synonymous (同义的).

□ We can use WordNet to specify candidate word senses in Word Sense Disambiguation.

□ We can transform Word Sense Disambiguation into a sequence labeling problem and then solve it using a neural network.

3.2 Sentence Semantics (5 pt)

3.2.1 Syntax-Driven semantic parsing (2 pt)

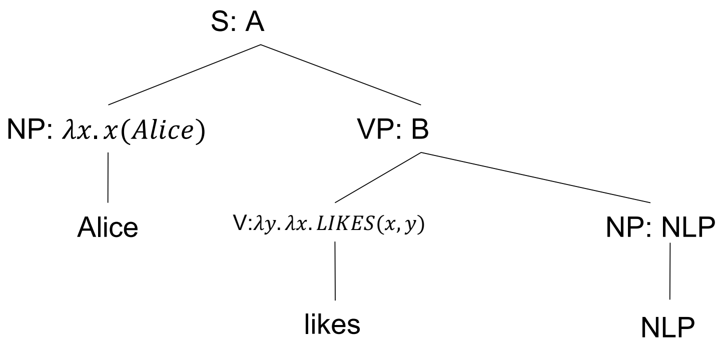
Given the syntax tree in Fig. 8, which is correct for both A and B in the figure after bottom-up reduction?

○ A: Likes(Alice, NLP) B: λx.Likes(x, NLP)

○ A: Likes(Alice, NLP) B: λx.Likes(NLP, x)

○ A: Likes(NLP, Alice) B: λx.Likes(x, NLP)

○ A: Likes(NLP, Alice) B: λx.Likes(NLP, x)



$$S \rightarrow NP\ VP\ \{NP.sem(VP.sem)\}$$
$$VP \rightarrow V\ \ NP\ \{V.sem(NP.sem)\}$$

Figure 8: Syntax-Driven Semantic Parsing

3.2.2 First-Order Logic (FOL) (3 pt)

Please choose all pair(s) in which the natural language (NL) sentence has consistent semantics with the first-order logic (FOL) expression.

☐ NL: All humans are mortal. FOL: $\forall x\,(Human(x) \rightarrow Mortal(x))$

☐ Some cats are black. FOL: $\exists x\,(Cat(x) \wedge Black(x))$

☐ NL: No dogs can fly. FOL: $\forall x\,(Dog(x) \rightarrow \neg CanFly(x))$

☐ NL: If it is raining, then people use umbrellas. FOL: $Raining \wedge \forall x\,(Person(x) \rightarrow UsesUmbrella(x))$

4 Discourse Analysis (5 pt)

4.1 Discourse Analysis and Coreference Resolution (3 pt)

Select all correct statement(s) below:

☐ Discourse parsing can be separated into two stages: elementary discourse units (EDU) segmentation and rhetorical structure theory (RST) parsing.

☐ In sentence "Her pay jumped to \$2.3 million" and "It is sunny", all noun phrases and pronouns should be detected as mentions.

☐ We can cast mention detection as binary classification in coreference resolution.

☐ Given two text spans in the same document "A: [Jane took a train from Paris to Istanbul]. B: [She had to attend a conference.]", B is the nucleus span and A is the satellite span.

4.2 Mention Ranking (2 pt)

Given a sentence "Can you can the can as a canner can it ?"(Chinese Translation: 你能像一个装罐工人一样装那个罐子吗？), please select the correct result of mention ranking in Figure 9.

○ A ○ B ○ C ○ D

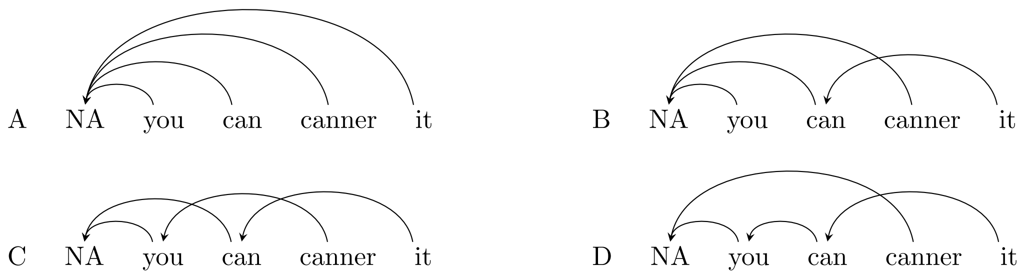


Figure 9: Mention Ranking

5 Information Extraction (5 pt)

5.1 Misc (3 pt)

Select all correct statement(s) about information extraction below:

- ☐ Any kind of named entity recognition tasks can be cast as a BIO sequence labeling task.
- ☐ Any kind of tasks in information extraction mentioned in this course can be cast as span/arc classification.
- ☐ In information extraction, partially-observed constituency trees can be used to predict spans of any kind (including flattened entities, nested entities, crossing entities, and so on).
- ☐ Given entity spans, predicting relations between them is just like predicting dependency arcs between words.

5.2 Partially-Observed Constituency Tree (2 pt)

Given a partially-observed constituency tree as in Fig. 10, one of the four observed nodes marked with A, B, C and D is in conflict with another node. Please find the node.

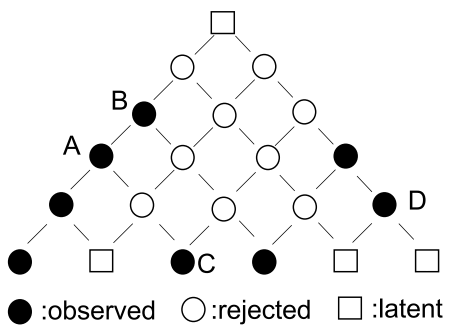


Figure 10: Partially-observed Constituency Tree

○ A ○ B ○ C ○ D