START OF QUIZ Student ID: 92642370,Chen,Tiffany

Question 1

Topic: Lecture 7
Source: Lecture 7

Briefly describe how dependency parsing differs from constituency parsing. (1)

Question 2

Topic: Lecture 6
Source: Lecture 6

Describe the purpose of the dot. (1)

Question 3

Topic: Lecture 8
Source: Lecture 8

What information do you think the word tokens on the stack/buffer are providing to the ML SR parser? (1)

Question 4

Topic: Lecture 6
Source: Lecture 6

Briefly describe the role of the scanner, predictor, and completer in the Earley Parser. (1)

Question 5

Topic: Lecture 5
Source: Lecture 5

What is fenceposting? Give two reasons we need it in the CYK algorithm. (1)

Question 6

Topic: Lecture 8
Source: Lecture 8

Imagine that we have a dependency parser that has a very good UAS (90+), but a very bad LAS (50-). Do you think that we could use the output of this parser as input to a neural translation model as is, or do you think that we should first re-train the labeling part of the algorithm to increase LAS? Doing both is probably the best solution, but I'm asking if you think that we could use the output of the existing model, even as we try to improve the quality of the labels. Explain. (2)

Question 7

Topic: Lecture 5
Source: Lecture 5

Let's say we wanted to modify PARSEVAL to take ambiguity into account. How might we use a PCFG and two gold references to account for ambiguous parsing? (2)

Question 8

Topic: Lecture 7
Source: Lecture 7

A deque is a data structure that mimics the operations of both a stack and a queue (ie, items can be added or removed to either end - check your 512 notes!). Do you think this data structure would be sufficient to replace the stack and buffer from SR parsing? Justify your answer. (2)

Question 9

Topic: Long
Source: Lecture 8

Let's say you have a friend who is developing a constructed language (conlang) for the epic fantasy novel he is writing, but the only language she knows is English, and she is just doing a word-for-word translation of English into this constructed language. She has no real knowledge of syntax, and has only ever thought that ”some words come after other words” (she's thinking like a language model). How would you use treebanks and dependency parsers to demonstrate to her that there is a whole ”hidden” structure that language must follow, and how would you help her develop a realistic grammar for her conlang? Do you think it would make more sense to build the grammar for a ”modified English”, and then do word-for-word translation, or to translate the words from English, and then build the grammar in the conlang? (3)

END OF QUIZ