START OF QUIZ
Student ID:
58115197,Zhang,Miaolin

Question 1

Topic: Lecture 4
Source: Lecture 4

Transition-based segmentation is very similar to the SR parser we saw last block, except it uses 2 FIFO structures, and doesn't require a stack. What is different about segmentation so that it doesn't require a stack? (1)

Question 2

Topic: Lecture 3
Source: Lecture 3

What is the motivation behind BPE (ie, what is it maximizing / minimizing)? (1)

Question 3

Topic: Lecture 4
Source: Lecture 4

Do you think that we could do Chinese Word Segmentation in a bottom-up way like we do with BPE? Why might this work (or not)? (1)

Question 4

Topic: Lecture 1
Source: Lecture 1

Knowing what you know about parsing, describe how derivation could be considered syntax, instead of morphology. In other words, how might we parse derivations? (1)

Question 5

Topic: Lecture 1
Source: Lecture 1

English is often described as an "analytic language with some fusional properties". Describe what that means, with an example. (1)

Question 6

Topic: Lecture 3
Source: Lecture 3

In the lab, you compared BPE with a more linguistically-motivated segmentation scheme. Intrinsically, the supervised method performs much better, but typically, BPE and its cousins work much better down-stream. Why do you think that is, taking into account the differences between the two methods? (2)

Question 7

Topic: Lecture 2
Source: Lecture 2

As a thought experiment, how might we build a calculator using an FST? Imagine that the FST reads input on one side of the tape, and generates operations (that are carried out by an algorithm) on the output side. (2)

Question 8

Topic: Lecture 2
Source: Lecture 2

In some ways, Statistical Machine Translation (SMT) was similar to an FST modified by a re-ordering model (ie, each input word had a corresponding output translation, and then the words were re-ordered to fit a language model). These models have been supplanted by NMT. What shortcomings of FSTs do you think put a ceiling on SMT performance? (2)

Question 9

Topic: Long
Source: Lecture 3

There is an alternative to BPE that randomly "forgets" to merge together certain subword sequences when it is creating its vocabulary (for example, "forget" will occasionally be represented as "for-get", occasionally as "forget", occasionally as "for-g-et", etc. What impacts do you think this might have on the vocabulary and model performance? Secondly, do you think there is a different impact between forgetting early iteration, mid iteration, and late iteration merges? (3)

END OF QUIZ