START OF QUIZ
Student ID:
22039382,Amal,Cenith

Question 1

Topic: Lecture 1
Source: Lecture 1

Explain why it's harder to rank polarity for words than simply categorizing them as "positive", "negative", or "neutral". (2)

Question 2

Topic: Lecture 4
Source: Lecture 4

Describe what features of neural networks allow for the creation of recursive neural networks. (1)

Question 3

Topic: Lecture 3
Source: Lecture 3

Identify the aspect of the following sentence: The special effects in Oppenheimer are subtle, but effective. (1)

Question 4

Topic: Lecture 2
Source: Lecture 2

We know that most sentiment words are adjectives, and many intensifiers and shifters are adverbs. Given a list of polar words, what tools could we use to discover intensifiers (beyond POS taggers and regexes)? Briefly explain. (1)

Question 5

Topic: Lecture 2
Source: Lecture 2

One of the goals of embeddings is that similar words are close to each other, and unrelated words are far apart. If we are using embeddings in our sentiment analysis toolkit, explain why we can't just "flip the polarity" of words modified by a negator (ie, [0.1, 0.3, 0.5] -> [-0.1, -0.3, -0.5]) (2)

Question 6

Topic: Lecture 1
Source: Lecture 1

Describe the Pollyanna principle, and how it complicates sentiment analysis. (1)

Question 7

Topic: Lecture 4
Source: Lecture 4

What are the assumptions we are making when we are implementing when we are creating a multi-task learner? Why wouldn't we just use the extra labels as input features to a single task learner? Wouldn't that be simpler? (2)

Question 8

Topic: Lecture 3
Source: Lecture 3

Briefly explain why TF-IDF is insufficient for identifying domain-specific aspects. (1)

Question 9

Topic: Long
Source: Lecture 3

In class, we've discussed ways of obtaining and expanding polarity lexicons, but we didn't talk about how to identify PPIs / NPIs. Write out pseudocode (ie, codish-looking stuff) that dives through a corpus of sentiment annotated documents across multiple domains, and identifies "potential potential items". If you make any assumptions about the data, be sure to list them. Don't actually write the code - this should be a designed algorithm, not a runnable piece of code. (3)

END OF QUIZ