

START OF QUIZ
Student ID:
74756297,..,Mandeep

Question 1

Topic: Lecture 3

Source: Lecture 3

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

Question 2

Topic: Lecture 1

Source: Lecture 1

Why do we need to update polarity lexicons regularly (probably more regularly than other lexicons)? (1)

Question 3

Topic: Lecture 4

Source: Lecture 4

Describe the propagation of error, and how it relates to neural architectures. (1)

Question 4

Topic: Lecture 1

Source: Lecture 1

Why is sentiment so tightly bound with domain? (1)

Question 5

Topic: Lecture 2

Source: Lecture 2

Sarcasm and irony are very difficult to detect with sentiment analysis methods. Outside of machine learning methods that consider larger contexts, do you think there is any way of detecting them with purely lexical (ie, word-based) resources? Briefly explain. (2)

Question 6

Topic: Lecture 3

Source: Lecture 3

We mentioned in class that "but clauses" are intensifiers. Do you think all (or at least most) concessions work the same way (some other concession words are "although", "nevertheless", "nonetheless", "even though", "considering that")? Briefly explain why or why not. (2)

Question 7

Topic: Lecture 4

Source: Lecture 4

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

(1)

Question 8

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, $[0.1, 0.3, 0.5] \rightarrow [-0.1, -0.3, -0.5]$) (2)

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