Auto-correct and Minimum edit distance

Total points 10

1.

Question 1

Select all the advantages of one hot encodings when representing words.

1 / 1 point

- [x] They are simple.

Correct

This is correct.

- [ ] They could be huge vectors.

- [ ] The embedding has no meaning.

- [x] It has no implied ordering.

Correct

This is correct. Why might an implied ordering be bad?

2.

Question 2

When using machine learning to learn word embeddings from a data set derived from a corpus, which type of supervision is commonly used?

1 / 1 point

- ( ) Supervised learning

- ( ) Semi-supervised learning

- (•) Self-supervised learning

- ( ) Unsupervised learning

Correct

Correct. Whilst the training data set is not labelled (as in unsupervised learning), it does contain data to supervise the learning process. This combination of supervised and unsupervised learning is self-supervised learning.

3.

Question 3

Which one of the following statements is correct?

1 / 1 point

To learn word embeddings you only need a vocabulary and an embedding method.

Learning word embeddings using a machine learning model is unsupervised learning as the input data set is not labelled.

The objective of a machine learning model that learns word embeddings is to predict word embeddings.

The meaning of the words, as carried by the word embeddings, depends on the embedding approach.

Correct

Correct: the specifics of the task are what will ultimately define the meaning of the individual words, e.g. assuming that words that are surrounded by the same kinds of words have similar meaning.

4.

Question 4

Which one of the following statements is false?

1 / 1 point

word2vec-based models cannot create word embeddings for words they did not see in the corpus they were trained on.

You need to train a deep neural network to learn word embeddings.

ELMo may have different word embeddings for the word "stable" depending on the context.

d) You can use a pre-trained BERT model to learn word embeddings on a previously unseen corpus.

Correct

Correct, this statement is false. For example, word2vec uses shallow neural networks, and GloVe doesn't use neural networks at all.

5.

Question 5

Consider the corpus "A robot may not injure a human being or, through inaction, allow a human being to come to harm." and assume you are preparing data to train a CBOW model. Ignoring punctuation, for a context half-size of 3, what are the context words of the center word "inaction"?

1 / 1 point

“being or through allow a human”

“being inaction human”

“through inaction allow”

“being or through inaction allow a human”

Correct

Correct, the context words are 3 words to both the left and the right of the center word.

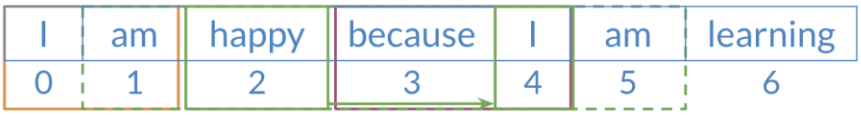
6.

Question 6

The code below is supposed to get the windows given words (some text) and a context size, C. It will return the context_words and the center_word. However there is a bug in this code.

Sliding window of words in Python

```
def get_windows(words, C):
    i = C
    while i < len(words):
        center_word = words[i]
        context_words = words[(i - C):i] + words[(i+1):(i+C+1)]
        yield context_words, center_word
        i += 1
```



1 / 1 point

The condition *i* < *len(words)* in the while loop is wrong and should be *i* < *len(words) - C.*

The condition *i* < *len(words)* in the while loop is wrong and should be *i* < *len(words) + C.*

We should start with i = 0 and not i = C.

context_words should be equal to *words[(i - C):i] + words[(i+1):(i+C)]* instead words[(i - C):i] + words[(i+1):(i+C+1)].

Correct

Yes, otherwise you will get an index out of bounds error.

7.

Question 7

Which one of the following statements is false?

1 / 1 point

Given the corpus "I think therefore I am", the word "you" cannot be represented.

The continuous bag-of-words model learns to predict context words given a center word.

Consider the corpus "A robot may not injure a human being or, through inaction, allow a human being to come to harm." and assume you are preparing data to train a CBOW model. Ignoring punctuation, for a context size of 3, the context words of the center word "inaction" are: "a", "allow", "being", "human", "or", and "through".

Given the corpus "I think therefore I am", the word "think" could be represented by the one-hot vector (1 0 0 0).

Correct

Correct. It's the reverse: the continuous bag-of-words model learns to predict a center word given context words. The continuous skip-gram model, presented in an earlier video, learns to predict context words given a center word.

8.

Question 8

You are designing a neural network for a CBOW model that will be trained on a corpus with a vocabulary of 8000 words. If you want it to learn 400-dimensional word embedding vectors, what should be the sizes of the input, hidden, and output layers?

1 / 1 point

8000 (input layer), 8000 (hidden layer), 400 (output layer)

400 (input layer), 400 (hidden layer), 8000 (output layer).

8000 (input layer), 400 (hidden layer), 8000 (output layer)

8000 (input layer), 400 (hidden layer), 400 (output layer)

Correct

Correct.

9.

Question 9

If you are designing a neural network for a CBOW model that will be trained on a corpus of 8000 words, and if you want it to learn 400-dimensional word embedding vectors, what should be the size of W1, the weighting matrix between the input layer and hidden layer, if it is fed training examples in batches of 16 examples represented by a 8000 row by 16 column matrix?

1 / 1 point

8000 rows by 16 columns

400 rows by 16 columns

400 rows by 8000 columns

16 rows by 8000 columns

Correct

Correct. The size of W1 does not depend on the batch size.

10.

Question 10

For a given training example, the output of a CBOW model is the vector $\hat{\mathbf{y}}$ below, predicting “happy” as the center word. The actual center word was “learning”. What is the cross-entropy loss for this example?

Note: log is logarithm base e.

$\hat{\mathbf{y}}$

0.083	am
0.03	because
0.611	happy
0.225	I
0.05	learning

$$J = -\sum_{k=1}^{V} y_k \log \hat{y}_k$$

1 / 1 point

0.49

1.49

2.49

2.99

Correct

Correct. The value in $\hat{\mathbf{y}}$ that corresponds to the actual word is 0.05, therefore J = -log 0.05 = 2.49