Started on	Wednesday, 8 January 2025, 2:29 PM
State	Finished
Completed on	Wednesday, 8 January 2025, 2:41 PM
Time taken	12 mins 7 secs
Grade	**4.00** out of 5.00 (**80**%)

Question 1

Correct

Mark 1.00 out of 1.00

Choose the correct statement/s concerning Word embeddings:

1. Word embeddings can be learned jointly with the main task you care about.

2. Word embeddings learned during one task can be utilized for other tasks as well similar to Transfer learning.

3. One of the ways to visualize word embeddings is by reducing the embedding vectors' dimension to 2D using PCA.

   ○   Only (2)

   ○   Both (1) & (2)

   ○   Only (3)

   ◉   All (1), (2) & (3)   ✔

Your answer is correct.

All statements are correct.

The correct answer is:
All (1), (2) & (3)

Question 2

Correct

Mark 1.00 out of

I want my NLP model to write poems like Shakespeare. What kind of task is this?

   ○   Machine translation task

1.00

○ Topic modeling task

○ Discriminative task

◉ Generative task ✔

Your answer is correct.

The correct answer is:
Generative task

Question **3**

Incorrect

Mark 0.00 out of
1.00

Consider a Text Vectorization layer which is adapted on a simple dataset as shown below:

```python
dataset = [
  "I like banana",
  "My favorite is apple",
]


text_vectorization = TextVectorization(output_mode="int", max_tokens=15,
output_sequence_length=10)


text_vectorization.adapt(dataset)


vocabulary = text_vectorization.get_vocabulary()
print(f"vocab = {vocabulary}")
```

```
vocab = ['', '[UNK]', 'my', 'like', 'is', 'i', 'favorite', 'banana', 'appl
```

What will the output be when the above text vectorization layer is applied to the test sentence below?

```
test_sentence = "I also like orange and apple"


encoded_sentence = text_vectorization(test_sentence)
print(f"encoded sentence = {encoded_sentence}")
```

○ encoded sentence = [5 0 3 0 0 8 1 1 1 1]

○ encoded sentence = [5 1 3 1 1 8 1 1 1 1]

○ encoded sentence = [5 1 3 1 1 8 0 0 0 0]

⦿ encoded sentence = [5 0 3 0 0 8 0 0 0 0]   ✖

Your answer is incorrect.

In the test sentence "I also like orange and apple", since the words 'also', 'orange', 'and' were

not present in the dataset, they will be assigned the unknown token index ([UNK]) i.e. 1.

Also, in the TextVectorization layer we have kept 'output_sequence_length = 10', so the

encoded sentence is padded with the empty string index i.e. 0 to have the same output length.

The correct answer is:
encoded sentence = [5 1 3 1 1 8 0 0 0 0]

Question **4**
Correct

Consider an Embedding layer defined below:

embedded = layers.Embedding(input_dim=20000, output_dim=256)

What will be the output shape when an input of batch size 32 and sequence length 600 is passed through this layer?

○ (32, 600, 256) ✔

○ (32, 600, 20000)

○ (20000, 600, 256)

○ (32, 20000, 256)

Your answer is correct.

In the Embedding layer it is mentioned output_dim=256, this means that embedding vectors of size 256 will be assigned to each token in the input sequence. So, if the input length is 600 then the output will be (600, 256). Considering the batch size of 32 then it will become (32, 600, 256).

The correct answer is:
(32, 600, 256)

Correct

Mark 1.00 out of 1.00

Question **5**

Correct

Mark 1.00 out of 1.00

Match the following:

Column I	Column II
a. Stop words	i. Text pre-processing technique used to break a word dow to its root meaning or word
b. Word Embeddings	ii. Words that do not add much meaning to a sentence
c. Lemmatization	iii. Splitting text into units – can be characters, words, grou sentences, etc

d. Tokenization	iv. Vector representation of words that map human languag into a structured geometric space

○ a-i, b-iv, c-ii, d-iii

◉ a-ii, b-iv, c-i, d-iii ✔

○ a-ii, b-iii, c-i, d-iv

○ a-i, b-iii, c-ii, d-iv

Your answer is correct.

● Stop words: Words that do not add much meaning to a sentence. They can safely be ignored without sacrificing the meaning of the sentence.

● Word Embeddings: Vector representation of words that map human language into a structured geometric space

● Lemmatization: Text pre-processing technique used to break a word down to its root meaning or word

● Tokenization: Splitting text into units – can be character, words, groups, sentences etc

The correct answer is:
a-ii, b-iv, c-i, d-iii