Grade 10 Computing

Unit 7 · How Large Language Models Work

WORKSHEET

Full marks 30

Name _____ Class _____ Roll no. ____ Date _____ Marks _____ / 30

1 Key words [4]

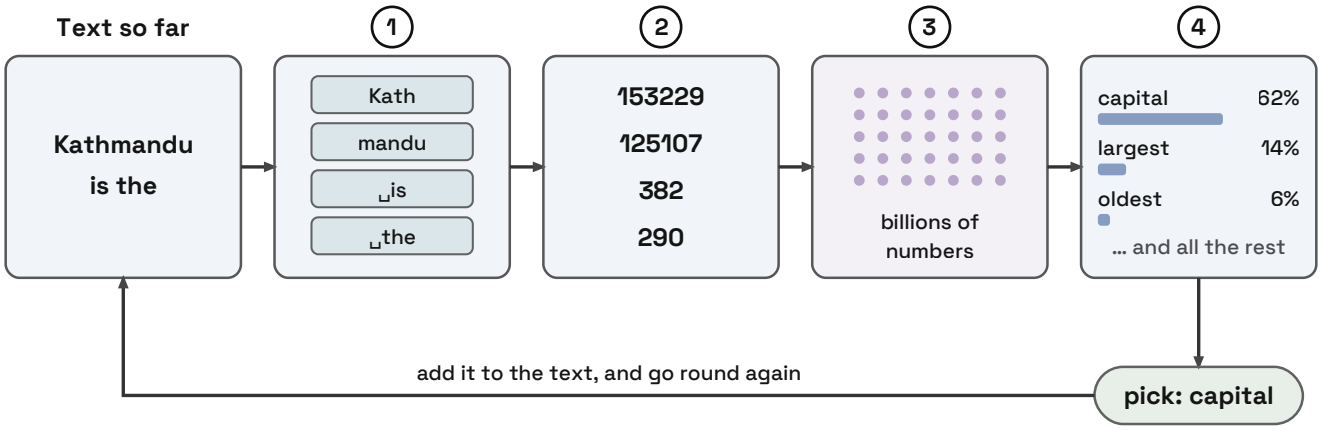
Fill in the blanks with words from the box. Two words in the box are not needed. Each blank carries 0.5 mark.

context window	**embedding**	**fine-tuning**	**grounding**	**hallucination**	**parameters**
pretraining	**temperature**	**tokenizer**	**vocabulary**		

The program that cuts text into tokens is a model's _____, and the list of every token it knows is its _____. The amount of text, in tokens, that a model can look at in one go is its _____. The setting that decides how a model picks the next token, always the most likely at 0, is its _____. The billions of numbers inside a model, learned in training, are its _____. Guessing the next token of trillions of tokens of text is the first stage of training, _____. Training an already-trained model further, on chosen examples, so that it gets better at one kind of task, is _____. An answer a model makes up that sounds true but isn't is a _____.

2 Label the diagram [2]

The diagram shows how a large language model writes. Write what belongs in place of 1 to 4. Each carries 0.5 mark.



1. _____ 2. _____

3. _____ 4. _____

3 Multiple choice [5]

Choose the correct option for each of the following questions.

i. Why don't language models use one token for every word? [1]

a. Words are too short to be useful.

b. The vocabulary would be enormous, and any word not in it couldn't be read at all.

c. Computers can't store words, only letters.

d. Tokens must always be single letters.

ii. A chatbot is asked the same question twice and gives two different answers. Why? [1]

a. It learned something new between the two questions.

b. One of the answers was copied from the internet.

c. It picks each token by chance, weighted by the chances it worked out, so the choices can differ each time.

d. Its tokenizer cut the question differently the second time.

iii. What does a **base model**, straight after pretraining, do best? [1]

a. answer questions politely and refuse harmful requests

b. carry on any text in the style of what it was given

c. look up facts in a database

d. remember every conversation it has had

iv. A chatbot gives a wrong founding year for a small school in Dolpa, in a confident sentence. What is the most likely reason? [1]

a. It was programmed to lie about small schools.

b. Its tokenizer couldn't read the word Dolpa.

c. The fact was rare or missing in its training text, and it wrote a likely-looking year anyway.

d. Its temperature was set to 0.

v. A chatbot is tested on 100 questions. It is right on 45 of 50 about the world and 25 of 50 about Nepal. Which statement is best? [1]

a. Its accuracy is 70%, so it is fairly good for everyone.

b. Its accuracy is 90% about the world but only 50% about Nepal, so it is biased against questions about Nepal.

c. Its accuracy is 50%, because Nepal is the lower score.

d. Nothing can be said from 100 questions.

4 What does it print? [4]

This program is a tiny counting model that looks back one word. Write exactly what it prints, in the box under it. Then answer ii.

```
1  words = "Ram is tall . Sita is kind . Ram is kind .".split()
2  follows = {}
3  for i in range(len(words) - 1):
4      w = words[i]
5      if w not in follows:
6          follows[w] = []
7      follows[w].append(words[i + 1])
8  print(follows["is"])
9  print(follows["Ram"])
10 print(len(follows))
```

```
OUTPUT
```

ii. If the model picks the word after “is” with `random.choice(follows["is"])`, what is the chance that it picks “kind”?

5 Trace the program [3]

This is the fixed `split_bill` from the unit. Fill in the trace table for `split_bill(23, 5)`: the first row after line 6, then one row for each time round the loop. Then write what line 11 prints.

```
1  def split_bill(total, people):
2      if people <= 0:
3          return []
4      share = total // people
5      left = total % people
6      shares = [share] * people
7      for i in range(left):
8          shares[i] = shares[i] + 1
9      return shares
10
11 print(split_bill(23, 5))
```

share	left	i	shares

```
OUTPUT
```

6 Check the code against its requirement [4]

Read the requirement and the program, then answer.

An AI wrote this function. The requirement: "`is_valid_mark(m)` returns True if a mark is a whole number from 0 to 100, both included, and False otherwise."

```
1  def is_valid_mark(m):
2      return m > 0 and m < 100
3
4  for m in [50, 0, 100, -5, 101, 72.5]:
5      print(m, is_valid_mark(m))
```

i. Write what the program prints.

```
OUTPUT
```

ii. Which of the six tests show a bug? Write the corrected line 2.

7 Guess, or say "I don't know"? [3]

Work it out in the space below.

A model takes a test of 100 questions, each with 4 choices. It knows 52 of the answers. For each of the other 48 it can guess, and gets exactly 1 guess in 4 right, or say "I don't know". Work out its score if it guesses and if it is honest: i. when a wrong answer scores 0; ii. when a wrong answer loses a mark. Which way of marking teaches a model to guess?

Answer:

8 Short answer [3]

Write a short answer to the following question.

For each mistake, say how much harm it could do (little, some or serious), and why:

i. A chatbot gives the wrong year for the first Everest summit in a class quiz for fun.
ii. A chatbot tells a farmer to mix twice the safe amount of a pesticide.
iii. A chatbot gives a made-up website as the source in a student's project.

9 Think [2]

Think, then answer.

Some people say "please" and "thank you" to a chatbot; others say it is pointless, since it feels nothing. Give one reason for each side. Does it matter, and for whom?

10 Activity: Make a tokenizer (not marked)

At a computer or phone, open the unit's page (the QR code) and find the activity **Make a tokenizer**. Learn from the English text first, then from English and Nepali. Fill in the table.

Learning from	Merges	Tokens in the English line	Tokens in the Nepali line
English only	none yet		
English only	until it stops		
English and Nepali	until it stops		