Room Number LT106

Seat Number 9

THE UNIVERSITY OF SYDNEY

Student Number 490051481

ANONYMOUSLY MARKED

(Please do not write your name on this exam paper)

CONFIDENTIAL EXAM PAPER

This paper is not to be removed from the exam venue

Computer Science

EXAMINATION

Semester 1 - Final, 2025

COMP4446/COMP5046 Natural Language Processing

EXAM WRITING TIME: 2 hours

READING TIME: 10 minutes

EXAM CONDITIONS:
This is a RESTRICTED OPEN book exam - specified materials permitted

MATERIALS PERMITTED IN THE EXAM VENUE:
(No electronic aids are permitted e.g. laptops, phones, calculators)

Formula sheet (provided in the exam paper by unit coordinator)

One A4 sheet of handwritten and/or typed notes double-sided

Bilingual dictionary (must have been pre-approved, as indicated by an official University of Sydney stamp)

MATERIALS TO BE SUPPLIED TO STUDENTS:
None

INSTRUCTIONS TO STUDENTS:
This exam consists of three sections (A: Multiple Choice Questions, B: Short Answer Questions, C: Programming Questions). All sections should be answered on this paper. Please use blue or black ink. If you need additional writing space, please use the extra pages provided at the end of this exam booklet. Only pages in this exam booklet will be marked.

Section A consists of 10 Multiple Choice Questions worth a total of 10 marks.

Section B consists of 20 Short Answer Questions worth a total of 40 marks.

Section C consists of 2 Programming Questions worth a total of 10 marks.

Please tick the box to confirm that your examination paper is complete (22 pages). ☑

This page is intentionally left blank.

Student Number:

Complete this on every page so we can find pages if they get separated during scanning.

Equations

Perplexity:

$$P(w_1, w_2 ... w_N)^{-\frac{1}{N}}$$

Layer normalization:

$$\mu = \frac{1}{d}\sum_{j=1}^{d} x_j$$

$$\sigma = \frac{1}{d}\sum_{j=1}^{d} (x_j - \mu)^2$$

$$y_i = \frac{x_i - \mu}{\sqrt{\sigma + \epsilon}} * \gamma + \beta$$

Self-attention with a dot product (assuming any changes to account for position have already been applied):

$$\begin{aligned}
\mathbf{q_i} &= Q\mathbf{x_i} \\
\mathbf{k_i} &= K\mathbf{x_i} \\
\mathbf{v_i} &= V\mathbf{x_i} \\
e_{ij} &= \mathbf{q_i}^\top \mathbf{k_j} \\
\alpha_{ij} &= \text{softmax}(e_{ij}) \\
\mathbf{t_i} &= \sum_j \alpha_{ij}\mathbf{v_j} \\
\mathbf{o_i} &= W_2\text{ReLU}(W_1\mathbf{t_i} + \mathbf{b_1}) + \mathbf{b_2}
\end{aligned}$$

Variants of attention:

Dot product

$$\mathbf{e} = \mathbf{s}^\top\mathbf{h}$$

Scaled dot product

$$\mathbf{e} = \frac{\mathbf{s}^\top\mathbf{h}}{\sqrt{d_h}}$$

Multiplicative / Bilinear

$$\mathbf{e} = \mathbf{s}^\top W\mathbf{h}$$

Reduced-rank multiplicative

$$\mathbf{e} = \mathbf{s}^\top(\mathbf{U}^\top\mathbf{V})\mathbf{h}$$

Additive / Feedforward

$$\mathbf{e} = \mathbf{b}\tanh(W_1\mathbf{h} + W_2\mathbf{s})$$

Non-linearities:

$$\begin{aligned}
\text{ReLU} &= \max(0, x) \\
\tanh &= \frac{e^x - e^{-x}}{e^x + e^{-x}} \\
\sigma &= \frac{a}{1 + e^{-x}}
\end{aligned}$$

Metrics:

$$\begin{aligned}
\text{Precision} &= \frac{TP}{TP + FP} \\
\text{Recall} &= \frac{TP}{TP + FN} \\
\text{Accuracy} &= \frac{TP + TN}{TP + TN + FP + FN} \\
\text{F-Score} &= \frac{2 * P * R}{P + R} = \frac{2TP}{2TP + FP + FN} \\
\text{F}_\beta\text{-Score} &= \frac{(1 + \beta^2)TP}{(1 + \beta^2)TP + \beta^2 FP + FN}
\end{aligned}$$

Cohen's Kappa:

$$\begin{aligned}
\kappa &= \frac{p_o - p_e}{1 - p_e} \\
p_o &= \frac{|\text{items with the same label}|}{N} \\
p_e &= \sum_{l \in labels} \prod_{a \in annotators} \frac{n_{la}}{N}
\end{aligned}$$

TF-IDF:

$$\begin{aligned}
\text{tf}_{t,d} &= \begin{cases} 1 + \log_{10}\text{count(t, d)} & \text{if count(t, d)} > 0 \\ 0 & \text{otherwise} \end{cases} \\
\text{idf}_t &= \log_{10}\left(\frac{N}{df_t}\right) \\
\text{tf-idf}_{t,d} &= tf_{t,d} * idf_t
\end{aligned}$$

This page is intentionally left blank.

Student Number: 49005148I

Complete this on every page so we can find pages if they get separated during scanning.

Multiple Choice Questions

Complete the answers below by completely filling in circles / squares next to the option(s) you are selecting. If the choices have ○ then select exactly one option. If the choices have □, select all correct options. Indicate your answer by filling the shape, e.g., ●. If you make a mistake, draw an X over your answer, e.g., ☒.

✓ 1. (1 mark) Which of these ways of storing a bag of words model is the **SLOWEST** to compute similarity? Select only one option.

- ● A vector, with a count for each word in the vocabulary ✓ one-hot-code
- ○ A sparse vector, with (token, count) pairs
- ○ A map, with tokens as keys and counts as values

✓ 2. (1 mark) What is the most significant difference between a static embedding and a contextual embedding? Select only one option.

- ● The contextual embedding of a word will vary across sentences ✓
- ○ Contextual embeddings require more computation to use
- ○ Static embeddings require fewer resources to train / create
- ○ Static embeddings were invented first

✗ 3. (1 mark) Which of these model types can easily do generation tasks, e.g., summarisation? Select all correct options.

- □ Encoder-only
- ✓ ☒ Encoder-decoder ✗ in lecture it says decoder-only is prefered
- ■ Decoder-only ✓

✓ 4. (1 mark) Which of the following is the main purpose of a Model? Select only one option.

- ● It calculates the score of an (input, output) pair ✓
- ○ It finds the correct output
- ○ It calculates the score for an input
- ○ It finds a high scoring output

✗ 5. (1 mark) Which of the following are true of top-1 and top-K sampling? Select all correct options.

- ✗ ■ They both use randomness to choose the output. ✓
- □ They sample differently. ✗
- □ Neither one considers the probability distirbution. ✗
- ■ They are both greedy methods. ✓

✓ 6. (1 mark) Which of these parts of the transformer help make training smoother? Select all correct options.

- ■ Residual connections ✓
- ■ Layer normalisation ✓
- □ Positional encoding
- □ Feedforward layers
- □ Self-attention
- □ Cross-attention

✓ ✗ 7. (1 mark) Which of these are benefits of using tools designed for annotation rather than using a spreadsheet or word processor? Select all correct options.

- ■ Improving consistency between annotators ✓
- □ Improving consistency across data
- ■ Improve speed of annotation ✓
- □ Reduce computational needs (memory, CPU power), needed
- ■ Make it easier to get started

✓ ✗ 8. (1 mark) Which of the following are significant advantages of retrieval augmented generation (RAG)? Select all correct options.

- ■ Responses are based on a source that can be shown to the user
- ■ Answers can update over time without retraining the language model ✓
- □ Prevents the language model from making any errors ✗
- ■ Switches memory usage from languge model parameters to vector database stoage

✓ ✗ 9. (1 mark) How do the Viterbi algorithm and CKY algorithm differ? Select all correct options.

- ■ Viterbi produces one output per word while CKY produces a structure with relationships between words HT ✓
- □ For n words, Viterbi has complexity O(n^2) while CKY has complexity O(n^3) ✗
- ■ Viterbi does not consider the sentence context for predicting a label but CKY does
- ■ Viterbi involves one pass to calculate probabilities and a second to extract the answer while CKY uses just one pass

? 10. (1 mark) When is the macro version of F-score more useful than the micro version? Selection all correct options.

-1

- ■ When there is a class imbalance and it is important to do well even on rare classes ✓
- ■ When some classes are harder than others and it is important to do well on all ?
- □ When we want to get as many examples right as possible ✗
- □ When the set of options does not contain an 'other' class ?

Student Number: 490051481

Complete this on every page so we can find pages if they get separated during scanning.

Short Answer Questions

In the questions below, please try to keep your answer inside the provided boxes. Marking will be done on scanned versions of the exams, so if you do need to go outside the box please keep your answer on the same page. Note, we have intentionally provided boxes that are much larger than necessary. Your answer does not need to fill the whole box.

✓ 11. (2 marks) Large language model agents designed with the ReAct approach combine two abilities. What are they?

Reasoning: happened before generating prediction

Acting: doing some thing beyond the generation. RAG is an example

✓ 12. (2 marks) In TF-IDF, the term frequency is a modified count of words. How is it modified and why?

It uses log for count of words in a document. It's modified because want to reduce the influence of very frequent words, like "the", "a"

✓ 13. (2 marks) Provide the BIO named entity tags for the sentence "Joe ate a chocolate croissant from Shadow Baking." assuming that the following entity types are available: Person (PER), Organisation (ORG), and Location (LOC).

Joe	B-PER
ate	O
a	O
chocolate	O
croissant	O
from	O
Shadow	B-ORG
Baking.	I-ORG

I don't know the meaning of this word, it looks like a organisation

14. What is one advantage and one disadvantage of greedy methods compared to exhaustive methods for inference?

✓ (a) (1 mark) Advantage (ie., a way greedy methods are better than exhaustive):

It does not cost a lot.

✓ (b) (1 mark) Disadvantage (ie., a way greedy methods are worse than exhaustive):

It might not be able always find the global optimum

15. Consider an encoder-decoder.

✓ (a) (1 mark) Where does the input to the **encoder** come from?

From User input sequence/data.

✓ (b) (1 mark) Where does the input to the **decoder** come from?

- From previous decoder's output
- First input is start token

Student Number: 49005148

Complete this on every page so we can find pages if they get separated during scanning.

16. Consider 'masked token prediction'.

(a) (1 mark) What is it?

✓

mask a token of a sentence, and predict the mask one base on rest of that sentence

✓ (b) (1 mark) What type of language model is it usually used to train?

Encoder, BERT

✓ 17. (1 mark) What is the purpose of instruction tuning for large language models?

Modify the model to do the task, including new task at inference time

18. Imagine you have been hired to make an AI bot to answer questions for COMP 4446 / 5046 next year.

(a) (1 mark) What data could you use for RAG and what filtering would you do to the data?

Data: ~~collect~~ from Ed and Canvas

Filter: remove those data which are not ~~addressed~~ resolved on Ed, not in English, not in comp 5046 outline

(b) (1 mark) Would you fine-tune an LLM as part of the system? Why / why not?

Yes, fine tuning can improve the capability on certain task (i.e. answer Comp 5046 question)

(c) (1 mark) What is a risk of providing this service?

It might not provide accurate answer and possibly mislead students.

Student Number: 49005148

Complete this on every page so we can find pages if they get separated during scanning.

✗ 19. (2 marks) The transofmrer can be viewed in terms of a residual stream. How is that different from the normal way of describing it?

It describe the residual connection .

✗ 20. (2 marks) How do Rotary Positional Embeddings allow self-attention to account for position?

If ~~two words~~ the distance between 2 words are same, them their similarity will be the same no matter where they are (location will not affect)

✓ 21. (2 marks) When providing examples for in-context learning, what way of ordering them would hurt performance?

If you order them according to the label, (i.e. put same label data together) it will hurt. Namely, unsorted example are best.

✓ 22. (2 marks) When annotating some data, what are two possible causes of low agreement?

- Unclear instruction/guidlines
- ambiguous data

23. (2 marks) What is the key difference between Direct Preference Optimisation (DPO) and Reinforcement Learning from Human Feedback (RLHF)?

DPO does not have a reward model
RLHF have a reward model

24. (2 marks) Given an example of an application where a system with high precision may not be useful and explain why.

Detect dangerous patients. Since the system ~~wan~~ does not want to miss any dangerous patients, otherwise they probably died because of don't get detected. So high Precision is not important than high recall. If you get high P and low R, this model can still not be regard as successful.

25. (2 marks) In an RNN, what is the purpose of the hidden vector?

Provide information about long-distance dependencies. Such as a later word want to find the relationship with earlier word. Calculate weighted avg for latter attention to use.

26. (2 marks) Attention uses queries, keys, and values. What would happen if all of the keys were set to be a vector of all 1s?

~~Query will always be 1 (since only 1 exist)~~
Since keys are all the same, so each query have same weight avg in ~~their value~~ t_i (t_i refer to formula sheet)

Student Number: 49005148

Complete this on every page so we can find pages if they get separated during scanning.

27. (2 marks) What are two key benefits of chrF over BLEU?

- It have higher tolerence to Typo
- It will not have 0 score when no 4-~~gr~~gram does not match (i.e. doesnot need multiple sentence)

In the next few questions, you will consider some pieces of code and answer questions about them. When asked for the purpose of the code, your answer should describe the goal of the person who wrote the code, not describe what each line does.

28. (2 marks) Consider the code below:

```
from pinecone import Pinecone
pc = Pinecone(api_key=pinecone_api_key)
index_name = 'semantic-search-fast'
pc.create_index(
    index_name,
    dimension=384,
    metric='dotproduct',
    spec=spec
)
while not pc.describe_index(index_name).status['ready']:
    time.sleep(1)
index = pc.Index(index_name)
for batch in dataset.iter_documents(batch_size=500):
    index.upsert(batch)
```

What is the purpose of this code?

Use Pinecorn as an external resources. Iterate the dataset with 500 batch size to find similar Vector in PineCorn. It's a kind of RAG

Student Number: 490051481

Complete this on every page so we can find pages if they get separated during scanning.

29. (2 marks) Consider the code below:

```
1  class RNN(nn.Module):
2      def __init__(self, input_size, hidden_size, output_size):
3          super(RNN, self).__init__()
4          self.hidden_size = hidden_size
5          self.i2h = nn.Linear(input_size, hidden_size)
6          self.h2h = nn.Linear(hidden_size, hidden_size)
7          self.h2o = nn.Linear(hidden_size, output_size)
8          self.softmax = nn.LogSoftmax(dim=1)
9          self.init_weights()
10
11     def init_weights(self):
12         initrange = math.sqrt(1 / self.hidden_size)
13         self.i2h.weight.data.uniform_(-initrange, initrange)
14         self.i2h.bias.data.zero_()
15         self.h2o.weight.data.uniform_(-initrange, initrange)
16         self.h2o.bias.data.zero_()
17         self.h2h.weight.data.uniform_(-initrange, initrange)
18         self.h2h.bias.data.zero_()
19
20     def initHidden(self):
21         return torch.zeros(1, self.hidden_size)
22
23     def forward(self, input_tensor, hidden):
24         new_hidden = torch.tanh(self.i2h(input_tensor) + self.h2h
               (hidden))
25         output = torch.tanh(self.h2o(new_hidden))
26         output = self.softmax(output)
27         return output, new_hidden                                ?
```

How would you change this model to use a ReLU non-linearity in the recurrent steps? Specify the line number and the change you would make.

in line . I will change "tanh" to "ReLU" Activation function 24

# Since you mean in "Recurrent Step", so I only include line 24. As line 25 is for output layer

30. (2 marks) Consider the code below:

```
1 import spacy
2 matcher = Matcher(nlp.vocab)
3 pattern = [{"POS": "ADJ"}, {"POS": "NOUN"}, {"POS": "NOUN", "OP":
      "?"}]
4 matcher.add("ADJ_NOUN_PATTERN", [pattern])
5 doc = nlp(text)
6 matches = matcher(doc)
7 for pattern_id, start, end in matches:
8     matched_span = doc[start:end]
9     print(matched_span.text, pattern_id, start, end)
```

What is the purpose of this code?

Specify the tag we need to find. Then find those ~~token~~ matched span which have those tags with that patter ~~(i.e. ADJ, NOUN, ...)~~ (i.e. pattern see line 3)

Student Number: 49005148

Complete this on every page so we can find pages if they get separated during scanning.

Programming Questions

In the next few questions, you will be given a task and a set of lines of code to do the task. Decide which lines to use and what order to place them in. Write the line numbers in order in the grids provided (one number in each box, in order from top to bottom). Note:

- If multiple orders are correct, we will accept all correct answers.
- You do not need to indicate indentation.
- Not all lines need to be used.
- There are extra pages at the back of the exam you can use to think.
- We provide more boxes than are needed.
- If you make a mistake, clearly put a line through the numbers and write a new response in the boxes.

31. (4 marks) Using the lines below, implement one step of Beam search. `beam` contains the current beam. `k` is the intended beam size. `token` is the current token. Provide your answer by writing the line numbers in the boxes to the right, in order, top to bottom.

```
  1  for label in labels:
✓ ~~2  new_beam.sort(reverse=True)~~
  ~~3  new_beam.sort()~~
  4  option = item + [label]
✓ ~~5  new_beam = []~~
  6  score = get_score(option)
  7  score = get_score(token)
  8  score = get_score(token, option)
  ~~9  beam = new_beam~~
✓ ~~10 beam = new_beam[:k]~~
  ~~11 for option in labels:~~ ✗
✓ 12 for score, item in beam:
✓ ~~13 new_beam.append((score, option))~~
  ~~14 new_beam.append(score)~~
  ~~15 new_beam.append(option)~~
```

5
12
1
4
8
13
2
10

5 new_beam=[],

12 For score, iten in beam

1 For label in labels

4 option = item+ labels

8 Score = get.score(token, option) # get score of token base on its label

13 new_bean. append ((score, option))

2 new-beam.sort(reverse=True)

10 beam = new-beam[:k]

For each beam we calculate the next possible score and add to new beam, choose top K

8

{word : score}

embeddings [w1].

32. (6 marks) Using the lines below, implement the word analogy task. a dictionary based object for storing a Bag of Words. embeddings is a list of word embeddings. w1, w2, and w3 are the IDs of words to use in the task. dim is the dimensionality of the embeddings.

i

1 2 3

1 2 3 4 6 8 ~~9~~ ~~9~~ 10 12 14 18 20

```
1  position = []
~~2  for i in range(dim):~~
3  position.append(embeddings[w1][i] - embeddings[w2
       ][i] + embeddings[w3][i])
4  length = math.sqrt(sum(p*p for p in position))
5  length = sum(p*p for p in position)
~~6  options = []~~
~~7  options = {}~~
~~8  for word, option in enumerate(embeddings):~~
9  if word in (w1, w2, w3): continue
10 olength = math.sqrt(sum(p*p for p in option))
~~11 olength = sum(p*p for p in option)~~
12 product = sum(option[i] * position[i] for i in
       range(dim))
13 score = sum(option[i] * position[i] for i in
       range(dim))
14 score = product / (length * olength)
15 options.append((score, word))
16 options.append(score)
17 options.append(word)
~~18 options.sort()~~
~~19 return options[0]~~
~~20 return options[-1]~~
```

1
2
~~4~~ 3
~~8~~ 4
6
8
9
10
12
14
18
20

```
options = []   6
For word, opt in enum(embedding)   8
    For i in range(dim)   2
        if word in (w1, w2, w3) : 9
            ~~continue~~
        Position = []   1
        position append ([w][i] ...)   3
        length 4
        olength 10
        Product 12
        Score 14
```

8

Student Number: 49005148

Complete this on every page so we can find pages if they get separated during scanning.

1
2
3
4
6
8
9
10
12
14
18
20

we only use w_1 w_2 w_3 in embedding

This page is left intentionally blank in case you need additional writing space.
Only pages that are stapled will be scanned. Scratch paper will not be scanned.

```
~~if word in (w1, w2, w3):~~
    ~~continue~~

Options = [ ]
For word, option in enumerate(embeddings):
    ~~For i in range(dim)~~
        ~~Position = [ ]~~
        ~~Position . append (embeddings[w1][i] - ... + ...)~~
    ~~length = √sum(p×p for p in position)~~
    olength = √sum(p×p for p in option)

    ~~product =~~
    Product = sum[option [i] × position[i] for i in range(dim))
    Score = Product / length × olength
    options . appen ((score , word))

options . sort( )
return options[-1]
```

This page is left intentionally blank in case you need additional writing space. Only pages that are stapled will be scanned. Scratch paper will not be scanned.

Student Number: 49005148

Complete this on every page so we can find pages if they get separated during scanning.

This page is left intentionally blank in case you need additional writing space. Only pages that are stapled will be scanned. Scratch paper will not be scanned.

END OF EXAMINATION