TÉCNICO LISBOA

Deep Learning

MSc in Computer Science and Engineering

MSc in Electrical and Computer Engineering

Final exam — February 26, 2022

Version A

Name: **Student n.:**

Registered in: ☐ Deep Learning (DEEC) ☐ Deep Learning (DEI)

Instructions

- You have 120 minutes to complete the exam.
- Make sure that your test has a total of 14 pages and is not missing any sheets, then write your full name and student n. on this page (and your number in all others).
- The test has a total of 19 questions, with a maximum score of 100 points. The questions have different levels of difficulty. The point value of each question is provided next to the question number.
- Please provide your answer in the space below each question. If you make a mess, clearly indicate your answer.
- The exam is open book and open notes. You may use a calculator, but any other type of electronic or communication equipment is not allowed.
- Good luck.

Part 1	Part 2	Part 3, Pr. 1	Part 3, Pr. 2	Total
32 points	18 points	25 points	25 points	100 points

Student n.:

Part 1: Multiple Choice Questions (32 points)

In each of the following questions, indicate your answer by *checking a single option.*

1. (4 points) Which of the following is the derivative of the tanh activation function?
   - ☐ $1 - \tanh(s)^2$
   - ☐ $\tanh(s)(1 - \tanh(s))$
   - ☐ 1 if $s > 0$, 0 otherwise.
   - ☐ None of the above.

2. (4 points) Your model is overfitting. Which of these strategies could mitigate the problem?
   - ☐ Augment your training set with more labeled data.
   - ☐ Decrease regularization.
   - ☐ Increase the learning rate.
   - ☐ All of the above.

3. (4 points) A **transformer-based** sequence-to-sequence model translates an input sentence of M words into an output sentence with N words. How does the number of computational operations (algorithmic complexity) increase as a function of M and N?
   - ☐ $O(M + N)$
   - ☐ $O(MN)$
   - ☐ $O(\max(M, N)^2)$
   - ☐ $O(M^N)$

4. (4 points) Which of the following sentences is **true**?
   - ☐ Neural networks work well in practice because their loss function is convex.
   - ☐ Variational auto-encoders are an instance of a latent variable model.
   - ☐ Generative adversarial networks maximize a lower bound of the data log-likelihood.
   - ☐ Convolutional neural networks are equivariant to rotations and scalings.

5. (4 points) Consider a linear model used in a binary classification task, where the output corresponds to the probability of $y = +1$ given the input, and trained with a binary cross-entropy loss. Which of these statements is **false**?

   - □ The model corresponds to a logistic regression classifier.
   - □ The model is trained with the perceptron algorithm.
   - □ The decision boundary for the network is a hyperplane in feature space.
   - □ The network can be trained using stochastic gradient descent.

6. (4 points) Which of these statements is **false**?

   - □ A commonly used objective in GANs is

     $$\min_G \max_D \mathbb{E}_{\mathbb{P}_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{\mathbb{P}_{\boldsymbol{\theta}}(h)}[\log(1 - D(G(h)))],$$

     where G is the generator and D is the discriminator.
   - □ GPT-3 is an encoder-only model trained on a masked language modeling objective.
   - □ VAEs often suffer from posterior collapse.
   - □ A common ethical concern with existing deep learning systems is their bias against minority groups not well represented in their training data.

7. (4 points) The first layer in AlexNet can be specified by the following code line in Pytorch:

```
nn.Conv2d(3, 96, kernel_size=11, stride=4)
```

where the first two parameters correspond, respectively, to the number of input and output channels, and no padding is used. Suppose that the input images are $223 \times 223 \times 3$. What is the size of the output after the above convolutional layer?

   - □ $213 \times 213 \times 96$
   - □ $54 \times 54 \times 11$
   - □ $54 \times 54 \times 96$
   - □ None of the above.

8. (4 points) Suppose that a **max pooling** layer with a 2×2 kernel and stride of 1 receives the following input:

$$\boldsymbol{x}_{\text{in}} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}.$$

What is the output of the pooling layer?

   - □ $\boldsymbol{x}_{\text{out}} = \begin{bmatrix} 3 & 4 \end{bmatrix}$.
   - □ $\boldsymbol{x}_{\text{out}} = \begin{bmatrix} 1.5 & 2.5 \\ 4.5 & 5.5 \end{bmatrix}$.
   - □ $\boldsymbol{x}_{\text{out}} = \begin{bmatrix} 5 & 6 \end{bmatrix}$.
   - □ $\boldsymbol{x}_{\text{out}} = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix}$.

You can use this space as draft.

Part 2: Short Answer Questions (18 points)

Please provide **brief** answers (1-2 sentences) to the following questions.

1. (6 points) Explain which problem long-short term memories (LSTMs) try to solve and how they do it.

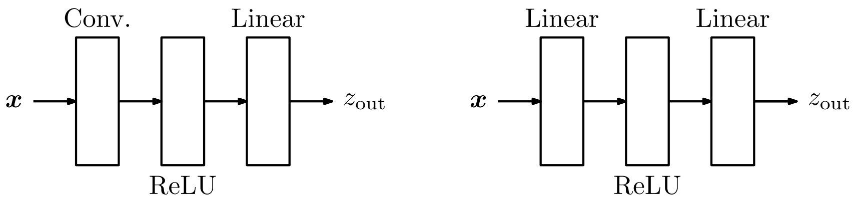
2. (6 points) What is an auto-encoder and why it can be useful?

3. (6 points) Explain why transformers need causal masking at training time.

Part 3: Problems (50 points)

Problem 1: Convolutional Neural Networks (25 points)

Consider the two networks depicted in Fig. 1. In the network of Fig. 1a the first block corresponds to a convolutional layer with a single 2×2 filter, no padding and stride $s = 1$. In the network of Fig. 1b the first block corresponds to a hidden layer with 4 units and ReLU activation.

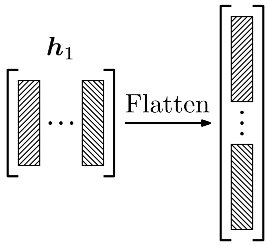


(a) Convolutional network. (b) Fully connected network.

Figure 1: Two network architectures to process the input $\boldsymbol{x}$.

In both networks we denote by $\boldsymbol{z}_1$ the input to the ReLU, by $\boldsymbol{h}_1$ the input to the rightmost linear layer, which in both cases comprises a single unit. We denote by z_{out} the scalar output, such that $\sigma(z_{\text{out}}) = \mathbb{P}[y = +1 \mid \boldsymbol{x}]$, where σ is the sigmoid function.

Note: Assume that, before the linear layer, $\boldsymbol{h}_1$ is flattened into a single column vector by stacking all columns of $\boldsymbol{h}_1$ together, as in the following diagram:



1. (5 points) Suppose that both networks expect as input a 3×3 matrix, which in the case of the network in Fig. 1b has been previously flattened. Fill in the table below. When counting the number of parameters in each network, make sure to consider the bias terms.

	Conv.	**Fully conn.**
Dimensions of $\boldsymbol{x}$	3×3	9×1
Dimensions of $\boldsymbol{z}_1$		
Dimensions of $\boldsymbol{h}_1$		
Dimensions of z_{out}	1	1
N. param. first layer	5	
N. param. last layer	5	

2. (8 points) Suppose that, after training, the parameters of the convolutional network in Fig. 1a are

$$\boldsymbol{K}_1 = \begin{bmatrix} -1 & -2 \\ 1 & 2 \end{bmatrix}, \qquad b_1 = 0 \qquad \boldsymbol{w}_{\text{out}} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \qquad b_{\text{out}} = -5,$$

where $\boldsymbol{K}_1$ and b_1 are the parameters of the convolutional layer, and $\boldsymbol{w}_{\text{out}}$ and b_{out} the parameters of the linear layer. Compute the output z_{out} of the network for the input

$$\boldsymbol{x} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}.$$

3. (8 points) Suppose that, after some training iterations, the parameters for the rightmost linear layer of the fully connected network in Fig. 1b are

$$\boldsymbol{w}_{\text{out}} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \qquad\qquad b_{\text{out}} = -5.$$

Let $(\boldsymbol{x}, y)$ be a sample in the dataset for which $y = +1$, and suppose that, when the input is $\boldsymbol{x}$, we have $\boldsymbol{h}_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix}^\top$. Perform one update of stochastic gradient descent to $\boldsymbol{w}_{\text{out}}$ and b_{out}, using a stepsize $\eta = 1$ and knowing that the loss considered is the negative log likelihood, i.e.,

$$L(z_{\text{out}}; y) = \begin{cases} -\log \sigma(z_{\text{out}}) & \text{if } y = +1; \\ -\log(1 - \sigma(z_{\text{out}})) & \text{if } y = -1. \end{cases}$$

Recall that $\sigma(z) = 1/(1 + \exp(-z))$ and $\sigma'(z) = \sigma(z)(1 - \sigma(z))$.

4. (4 points) Briefly explain why using a ReLU activation may be preferable over a sigmoid activation in very deep networks.

You can use this space as draft.

Student n.:

Problem 2: Sequence Classification (25 points)

Ada is developing a system for quote attribution, where the input is a sentence (a quote) and the output should be the author of that sentence, among three possibilities: Mark Twain (class 1), Albert Einstein (class 2), and GPT-3 (class 3). Ada trains an RNN-based classifier on a corpus of quotes, obtaining the following model parameters:

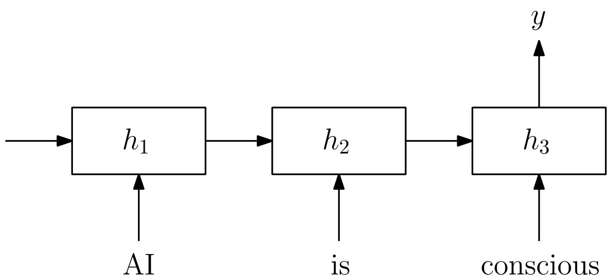
- The initial hidden state of the RNN, $\boldsymbol{h}_0$, is all-zeros.
- The input-to-hidden matrix and the hidden-to-output matrix are respectively:

$$\boldsymbol{W}_{hx} = \begin{bmatrix} 0 & -1 & 2 \\ 1 & -2 & 0 \end{bmatrix}, \quad \boldsymbol{W}_{yh} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \\ 2 & -1 \end{bmatrix}.$$

- The recurrent matrix $\boldsymbol{W}_{hh}$ is the identity matrix.
- All biases are vectors of zeros.
- The RNN uses `relu` activations.

She now wants to use her model to predict the author of the quote "AI is conscious". The relevant word embeddings are:

$$\boldsymbol{x}_{\text{AI}} = [0, 1, 1]^\top, \quad \boldsymbol{x}_{\text{is}} = [1, 0, -1]^\top, \quad \boldsymbol{x}_{\text{conscious}} = [0, -1, 0]^\top.$$



1. (8 points) Assume Ada's model uses the last state of the RNN ($\boldsymbol{h}_3$) to make the prediction. Who is the predicted author of the quote? Show all your calculations.

(you can continue in the next page)

2. (8 points) Ada realized that her model is much better if she uses a simple attention mechanism (instead of using the last state of the RNN) as the pooling strategy. She adds a query vector $\boldsymbol{q}$ as an extra model parameter, and trains the entire model, obtaining the same parameters as above and $\boldsymbol{q} = [2, -1]^\top$. This model uses as keys and values the RNN states $\boldsymbol{h}_1 = [1, 0]^\top$, $\boldsymbol{h}_2 = [0, 1]^\top$, $\boldsymbol{h}_3 = [1, 3]^\top$. Using **scaled dot product attention**, what is the new prediction for the same quote and which word receives the largest attention probability? Show all your calculations.

3. (4 points) Is there any possible input for which any of the two models above (with the given parameters) can assign probability higher than $\frac{1}{3}$ to Mark Twain? What if we replace relu by tanh activations? Justify your answer.

4. (5 points) Give one example of a transformer-based pretrained model that Ada could use for this task and the necessary steps to use it.

You can use this space as draft.

You can use this space as draft.