**Academic Integrity Notice**

**Strict Prohibition on AI-Generated Code**

**The use of ChatGPT, Copilot, or any other AI tools to generate code for this assignment is strictly prohibited. Such actions constitute academic misconduct. Any submission found to contain AI-generated code will receive a grade of 0. We employ advanced detection methods to identify AI-generated content. Additionally, instructors may assess your understanding by asking you to explain your code and reasoning. So be ready to answer any questions based on your homework.**

**Please note that AI-generated code often lacks the unique stylistic elements and logical progression characteristic of human-written code. Such discrepancies are easily detectable by experienced instructors and automated tools.**

**Maximum grade you can get is 100. E.g if you receive 110 your score will be floored down to 100.**

## **Part 1: Natural Language Processing (NLP)**

## **(50 points)**

### **Theory (Answer briefly):**

1. Explain what are embeddings?
2. What is the role of the attention mechanism in transformers?
3. Describe the main differences between BERT and GPT architectures.
4. Explain the difference between RNN, LSTM, GRU.

### **Practical (PyTorch + Hugging Face Transformers)**

**Task:** Fine-tune a pre-trained BERT model for sentiment classification on the IMDb dataset.

**Dataset:** IMDb movie reviews dataset (available in Hugging Face Datasets)

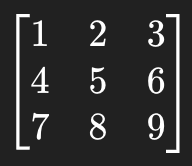
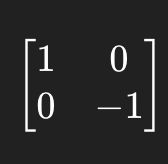
**Steps:**

1. Load the IMDb dataset using datasets library (huggingface/datasets).
2. Preprocess text data:  
   * Tokenize using BertTokenizer (from transformers library).
   * Pad and truncate sequences.
3. Prepare DataLoader with batching.
4. Load a pre-trained BERT base model for sequence classification and extract the embeddings from Bert model.
5. Once you have embeddings create a dataframe from those embeddings and treat the embeddings as your dataset now (e.g n columns, k rows)
6. Train a classifier (e.g Logistic Regression, Decision Tree Classifier, Random Forest and XGboost) for at least 5 epochs on the training split.
7. Evaluate on the validation set as well as your own data (you can make up a couple self-made sentences)
8. report accuracy, precision, f1, ROC curves. Explain the results with markdowns/comments
9. Save the trained model.

## **Part 2: Generative Models**

## **(50 points)**

**a Standard Convolution (Downsampling)**

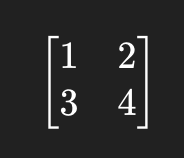
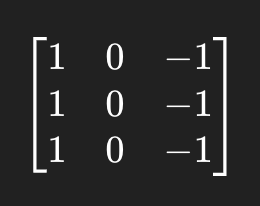
* Input:  
  
* Kernel:
*   
   Stride: 2
* Stride: 1
* Padding: 0

**b Compute Transposed Convolution (Upsampling/Deconvolution)**

### **Theory (Answer briefly):**

1. Compare GANs and VAEs. What are their main strengths and weaknesses?
2. Explain the concept of the diffusion process in diffusion models.
3. Describe the role of the discriminators and generators in a GAN. How are they interacting with each other? What do we need for training and what for inference?

**Given:**

* Input:  
  
* Kernel:
*   
   Stride: 2
* Padding: 0
* Output Padding: 0

### **Practical (PyTorch)**

**Task Details**

#### **1. Dataset: CelebA (CelebFaces Attributes Dataset)**

* Over 200,000 celebrity face images
* Aligned and cropped to faces
* Publicly available and widely used for GAN benchmarks

#### **2. Goal**

* Train a GAN network that learns to generate new human-like faces from random noise. You are free to choose your own architecture of GANs.

## **Part 3: Reinforcement Learning (RL)**

## **(15 bonus points)**

### **Theory (Answer briefly):**

1. **What is reinforcement learning? How is it different from supervised learning?**
2. **Define the core components of a reinforcement learning system for a self driving car.**

* Agent
* Environment
* State
* Action
* Reward
* Policy
* Value function
* Model of the environment (if any)