Winstars AI DS internship test

The goal of this test is to evaluate your skills and knowledge in Data Science and related fields. We propose to solve two tasks below that contain the exercises related to Machine Learning, Computer Vision, NLP, and regular coding. Both tasks require skills that will be useful in the projects you will work on in the company. Hope it will be interesting to you. In case of any issues or misunderstandings - contact us. Please follow the instructions and Good Luck!

General requirements for the test:

- The source code should be written in ***Python 3***.
- The code should be clear for understanding and well-commented.
- All solutions should be put into the ***GitHub*** repository. Each task should:
    - be in a ***separate folder***.
    - contain its own ***readme file*** with a solution explanation and details on how to set up the project.
    - ***requirements.txt*** with all libraries used in the solution.
- All the documentation, comments, and other text information around the project should be written in ***English***.
- Demo that should be represented like a ***Jupyter Notebook*** and contain examples of how your solution is working including a description of the edge cases.

Task 1. Image classification + OOP

In this task, you need to use a publicly available simple MNIST dataset and build 3 classification models around it. It should be the following models:

1) Random Forest;
2) Feed-Forward Neural Network;
3) Convolutional Neural Network;

Each model should be a separate class that implements ***MnistClassifierInterface*** with 2 abstract methods - ***train*** and ***predict***. Finally, each of your three models should be hidden under another MnistClassifier class. MnistClassifer takes an ***algorithm*** as an input parameter. Possible values for the algorithm are: ***cnn***, ***rf***, and ***nn*** for the three models described above.

The solution should contain:

- Interface for models called ***MnistClassifierInterface.***
- 3 classes (1 for each model) that implement ***MnistClassifierInterface.***
- ***MnistClassifier,*** which takes as an input parameter the name of the algorithm and provides predictions with exactly the same structure (inputs and outputs) not depending on the selected algorithm.

Task 2. Named entity recognition + image classification

In this task, you will work on building your ML pipeline that consists of 2 models responsible for totally different tasks. The main goal is to understand what the user is asking (NLP) and check if he is correct or not (Computer Vision).

You will need to:

- find or collect an animal ***classification/detection dataset*** that contains at least 10 classes of animals.
- train ***NER model*** for extracting animal titles from the text. Please use some ***transformer-based model (not LLM)***.
- Train the animal ***classification model*** on your dataset.
- ***Build a pipeline*** that takes as inputs the text message and the image.

In general, the flow should be the following:

1. The user provides a text similar to “There is a cow in the picture.” and an image that contains any animal.
2. Your pipeline should decide if it is true or not and provide a boolean value as the output. You should take care that the text input will not be the same as in the example, and the user can ask it in a different way.

The solution should contain:

- ***Jupyter notebook*** with exploratory data analysis of your dataset;
- Parametrized ***train and inference*** .py files for the NER model;
- Parametrized ***train and inference*** .py files for the Image Classification model;
- Python script for the entire pipeline that takes ***2 inputs (text and image) and provides 1 boolean value as an output***;