Data Science internship test

The goal of this test is to evaluate your skills and knowledge in such Data Science fields as Natural Language Processing and Computer Vision. We propose to solve 2 tasks linked with these Data Science domains. Both tasks require skills that will be useful in the projects you will work on in the company. We assume that for such a limited time it can be hard to achieve production-quality solutions, so you should also attach a short report with ideas on how to boost the performance of your solution.

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.
- Report with potential improvements in PDF.

Please take into account recommendations at the end of each task. It is not mandatory to follow them, but we assume that it can simplify the process of the test task completion.

Good luck, and have a nice time working on the task!

Task 1. Natural Language Processing. Named entity recognition

In this task, we need to train a named entity recognition (NER) model for the identification of mountain names inside the texts. For this purpose you need:

- Find / create a dataset with labeled mountains.
- Select the relevant architecture of the model for NER solving.
- Train / finetune the model.
- Prepare demo code / notebook of the inference results.

The output for this task should contain:

- Jupyter notebook that explains the process of the dataset creation.
- Dataset including all artifacts it consists of.
- Link to model weights.
- Python script (.py) for model training.
- Python script (.py) for model inference.
- Jupyter notebook with demo.

Recommendation:

- ***Look into possibilities of ChatGPT for dataset generation;***
- ***Check BERT-based pre-trained models for NER problem;***

Task 2. Computer vision. Sentinel-2 image matching

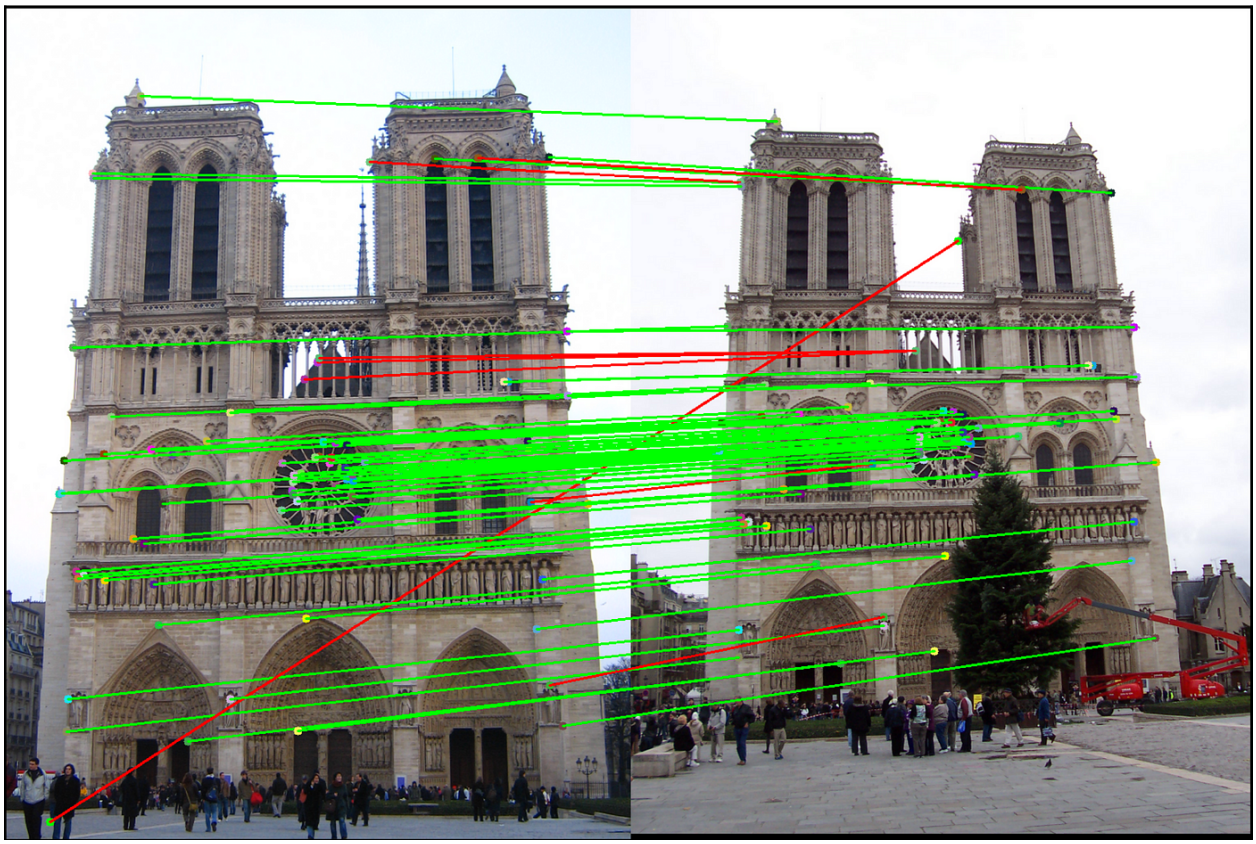
In this task, you will work on the algorithm (or model) for matching satellite images. For the dataset creation, you can download Sentinel-2 images from the official source here or use our dataset from Kaggle. Your algorithm should work with images from different seasons. For this purpose you need:

- Prepare a dataset for keypoints detection and image matching (in case of using the ML approach).
- Build / train the algorithm.
- Prepare demo code / notebook of the inference results.

The output for this task should contain:

- Jupyter notebook that explains the process of the dataset creation.
- Link to the dataset (Google Drive, etc.).
- Link to model weights.
- Python script (.py) for model training or algorithm creation.
- Python script (.py) for model inference.
- Jupyter notebook with demo.

Your demo notebook should have the functionality for observing detected keypoints and their matches in a way like this:



Recommendation:

- **Check deep learning possibilities to solve this task. Classical solutions can be not accurate enough for images from different seasons.**
- **Satellite images have large sizes. You should think about how to process them in order not to lose the quality.**
- **Some initial knowledge about satellite imagery processing you can find here.**