**Amazon NER Project**

This project is a Python notebook that uses the Flair framework to recognize named entities in texts about the Amazon. The project allows extracting relevant information about the environmental and social issues that affect this region, such as pollution, illegal mining and deforestation. The project also offers functions to train text classification models and generate summaries of the cases analyzed. The objective is to facilitate the analysis and monitoring of the situation of the Amazon.

**Installation**

To run this project, you need to have Python 3.6 or higher and Jupyter Notebook installed on your system. You also need to install the following libraries:

* Flair: A natural language processing framework for PyTorch. You can install it with pip install flair.
* Pandas: A data analysis and manipulation tool. You can install it with pip install pandas.
* NLTK: A natural language toolkit for Python. You can install it with pip install nltk.
* Sumy: A module for automatic summarization of texts. You can install it with pip install sumy.

**Usage**

To use this project, you need to download or clone this repository and open the notebook file amazon\_ner.ipynb in Jupyter Notebook. You can then run the cells in order or select the ones you are interested in. The notebook contains comments and explanations for each step of the process.

The main steps are:

* Loading and preprocessing the data: The data consists of texts extracted from news articles and reports about the Amazon. The preprocessing includes tokenization, lemmatization and removal of stopwords.
* Recognizing named entities: The project uses Flair’s pre-trained models to recognize named entities such as locations, organizations and persons in the texts. The recognized entities are stored in a Pandas dataframe for further analysis.
* Training text classification models: The project uses Flair’s text classifier to train models to classify the texts into categories such as environment, social, economy and politics. The models are evaluated on a test set and their performance metrics are displayed.
* Generating summaries: The project uses Sumy’s Luhn algorithm to generate summaries of the texts based on their most important sentences. The summaries are also stored in a Pandas dataframe for further analysis.

**INTEREST OBJECTS - IMAGES**

This project is a Python notebook that uses the Segment Anything Model (SAM) framework to segment geospatial images with natural language queries. The project allows selecting an area of interest on an interactive map and entering a keyword to display the segmented objects that match the keyword. The project also offers functions to download and merge map tiles into a single GeoTIFF file and to convert videos to frames for further analysis. The objective is to facilitate the analysis and monitoring of geospatial data.

**Installation**

To run this project, you need to have Python 3.6 or higher and Jupyter Notebook installed on your system. You also need to install the following libraries:

* Segment Geospatial: A library that adapts the code of SAM to work with geospatial data. You can install it with pip install segment-geospatial.
* Grounding Dino: A library that uses SAM’s pre-trained models to segment objects selected by keywords. You can install it with pip install groundingdino-py.
* Leafmap: A library for creating interactive maps with Python. You can install it with pip install leafmap.
* Local Tile Server: A library for serving local map tiles in Jupyter Notebook. You can install it with pip install localtileserver.

**Usage**

To use this project, you need to download or clone this repository and open the notebook file objetos\_de\_interes.ipynb in Jupyter Notebook. You can then run the cells in order or select the ones you are interested in. The notebook contains comments and explanations for each step of the process.

The main steps are:

* Loading and displaying the map: The project uses Leafmap’s map object to display an interactive map with various basemaps and layers. The map can be zoomed in and out and clicked to see the coordinates of any point.
* Selecting an area of interest: The project uses Leafmap’s draw control tool to allow the user to draw a rectangle on the map and select an area of interest. The coordinates of the rectangle are stored in a variable for later use.
* Downloading and merging map tiles: The project uses Local Tile Server’s tms\_to\_geotiff function to download and merge the map tiles in the selected area into a single GeoTIFF file. The file is saved in a folder called data and displayed on the map as a new layer.
* Segmenting objects by keywords: The project uses Grounding Dino’s LangSAM class to segment objects in the GeoTIFF file by keywords. The keywords can be entered by the user or selected from a list of predefined ones, such as tree, building, road, etc. The segmented objects are displayed on the map as polygons with different colors.
* Converting videos to frames: The project uses OpenCV’s video capture and frame extraction methods to convert videos stored in a folder called videos into frames. The frames are saved in a folder called images and can be used for further analysis.

**LICENCE**

This project is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. This means that:

* You are free to share (copy and redistribute) the material in any medium or format for non-commercial purposes only, as long as you give appropriate credit to the author and indicate if changes were made.
* You may not use the material for commercial purposes (such as selling, renting, licensing, or otherwise exploiting the material for monetary gain).
* You may not remix, transform, or build upon the material (such as modifying, adapting, or creating derivative works from the material).
* You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

You can view a copy of this license at <https://creativecommons.org/licenses/by-nc-nd/4.0/>.