Data Cleanisng

Description

This repository contains data cleansing scripts for various parts of the program. - Tabular data cleansing - Text validation - Data cleansing for Main and Electronics NLP-pipeline - Data cleansing for Address-Analytics

The cleansing processes includes Text validation, Armenian cleansing, English cleansing, Translation, Address cleansing, and tabular data cleansing. The data files are primarily in .xlsx and .csv format, and the project is capable of handling batch processing for efficient data management.

Features

The repository implements the following key steps: - **ValidatingText**: Validates text, to make sure that good names are interpretable. - **NLPCleasning**: Cleans,preprocesses and preapares data for NLP-pipeline - **TabularDataCleasning**: General cleasning for Tabular data - **AddressPreprocessing**: Processing Address information for address analytics.

Using a configuration file (`config.json`) you need to specify the input and ouptut data paths, as well as parameters if needed. You need to specify parameters in NLP-pipeline, and text validation.

Configuration

The `config.json` file contains the paths to the input data and configurable parameters for the preprocesing. You need to specify the input and ouptut data paths, as well as parameters if needed.

Data: Specifies paths to input data and ouput files. Params: Controls settings for - n_jobs for parallel processing - batch_size for handling batch operations. - input_column_name: cahnge columns name if the original column name is not a "GOOD_NAME". (This is the case for invoice and declaration data).

Inputs and Ouputs of program

Address

Input data name - `division information full.xlsx` Output data name - `address_output_data.csv` Output contains following columns * OIN * full_address

Text Validation

Input data name - `input.csv` -- could be any file containig good name Output data name - `output.csv` Output contains following columns * original good name column
* Is_Valid_Product_Name

NLP_Cleansing

Input data name - `input.csv` -- could be any file containig good name Output data name - `output.csv` Output contains following columns * original good name column
GOOD_NAME_CL
GOOD_NAME_CL_TR
GOOD_NAME_CL_TR2

GOOD_NAME_CL_EL
GOOD_NAME_CL_EL_TR
GOOD_NAME_CL_EL_TR2

All this columns are needed for NLP pipeline

Tabular Data Cleasning

Input data name - `input.csv` -- could be any tabular data Output data name - `output.csv` Output original data columns in cleaned form

Installation

Follow these steps to set up the project on your local machine:

1. **Clone the repository**:

```bash git clone https://github.com/MindwiseLLC/data-cleansing.git 2. **Ensure Python 3.11 is installed**:

This project requires Python 3.11. You can download and install it from the official Python website: https://www.python.org/downloads/.

1. **Create a virtual environment**

After installing Python 3.11, create a virtual environment for the project:

**For Windows:**

```
python -m venv data-prep
data-prep/Scripts/activate
```

**For macOS/Linux:**

```
python3.11 -m venv data-prep
source data-prep/bin/activate
```

4. **Install dependencies**

After activating the virtual environment, install the required dependencies:

```
pip install -r requirements.txt
```

5. **Run the main script**

First, navigate to the ***'code'*** directory using the command:

```
cd code
```

Second, choose the cleasning script you need.

To run the chosen script, execute the following command:
```

```
python your_chosen_script.py
```

This will execute the pipeline according to the settings defined in config.json.

Possible scripts are the following

```
python NLP_Cleansing.py
```

```
python address_cleansing.py
```

```
python tabular_data_cleansing.py
```

```
python text_validation.py
```