HARMONIZE

socio4health

Introduction, Functions and Applications

FIOCRUZ, Rio de Janeiro - November 2025

uniandes FIOCRUZ IAI Inter-American Institute for Global Change Research BSC INPE UNIVERSIDAD PERUANA CAYETANO HEREDIA INDOMET INSTITUTO DOMINICANO DE METEOROLOGÍA wellcome

Our Team

Diego Irreño

Erick Lozano

Juan Montenegro

Ingrid Mora

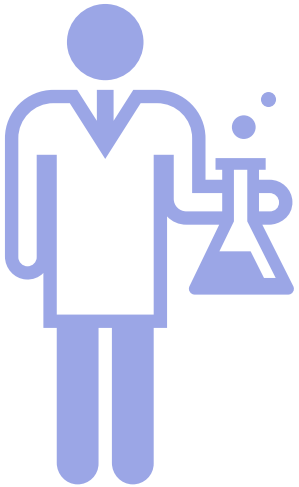
Felipe Aramburo

Mauricio Santos

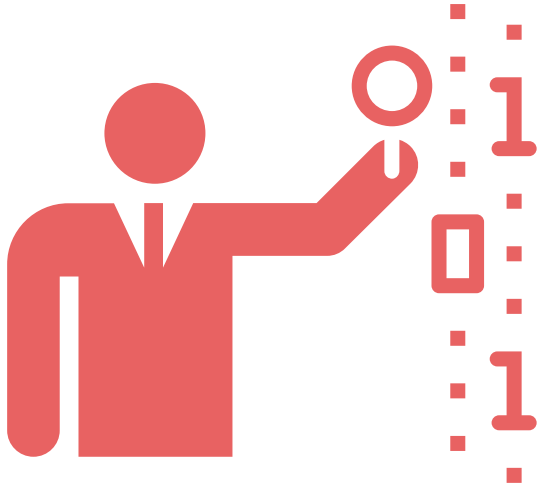
Objective

To simplify the complex process of **collecting** and **merging** data from multiple sources focusing on **sociodemographic** datasets from different countries, offering a solution that integrates and relates heterogeneous data in an **accessible** and **scalable** tool

Who should use socio4health?

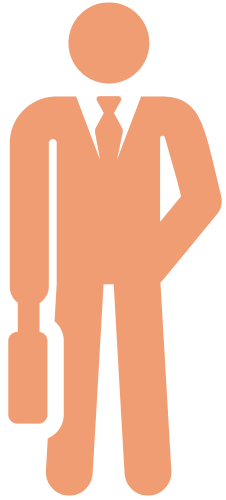


scientist & researchers



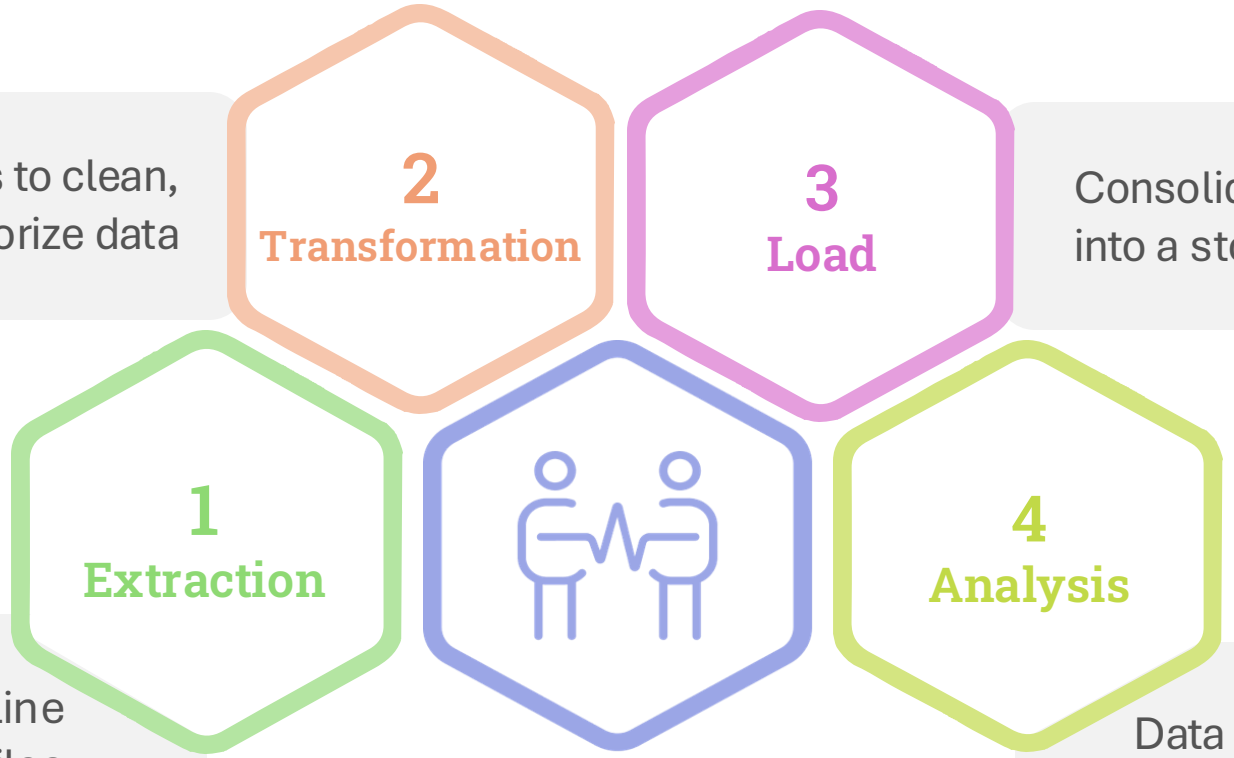
data scientist

developers



practitioners and decision makers

Socio4health: Workflow



Retrieve data from online sources or local files

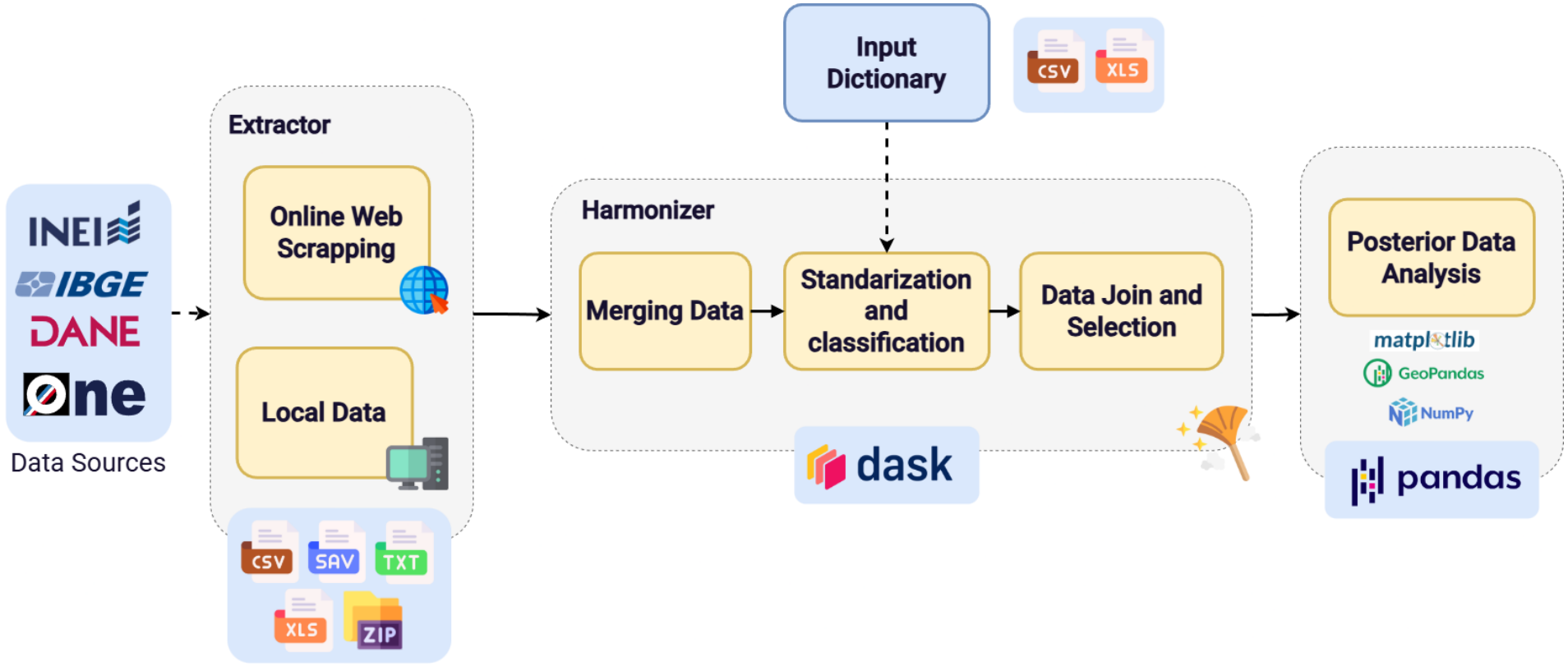
Apply transformations to clean, filter and categorize data

Consolidate transformed data into a storable dataset

Data can be processed by other powerful data analysis libraries

Achieve data governance over sociodemographic microdata

Socio4health: Structure



Functionalities

Extractor

Input

Raw sociodemografic microdata files

Extract

General parameters:

- **Input path:** URL or local directory
- **Output path:** Location of downloaded files
- **Download extension:** Needed file extensions (.zip, .csv, .txt, .shp, etc.)
- **Separator:** Separator used for reading CSV files
- **Encoding:** For reading files

Online extraction parameters:

- **Depth:** Levels of linked pages
- **Key words:** List of words to filter files when scrapping

FWF files parameters:

- **Column names:** Character vector
- **Column specifications:** Pairs of boundaries

Output

List of Dask dataframes

Harmonizer

Vertical Merge

- **Similarity threshold:** Join column names duplicates
- **Nan threshold:** Filter null columns

Data Selector

- **Key Column:** Column to be filtered.
- **Key Values:** Values to be filtered
- **Categories:** Mapped variables from the dictionary
- **Extra Columns:** Extra variables needed after filtering

Join Data

- **Join Key:** Column used as a relationship between DataFrames

Input

List of Dask dataframes

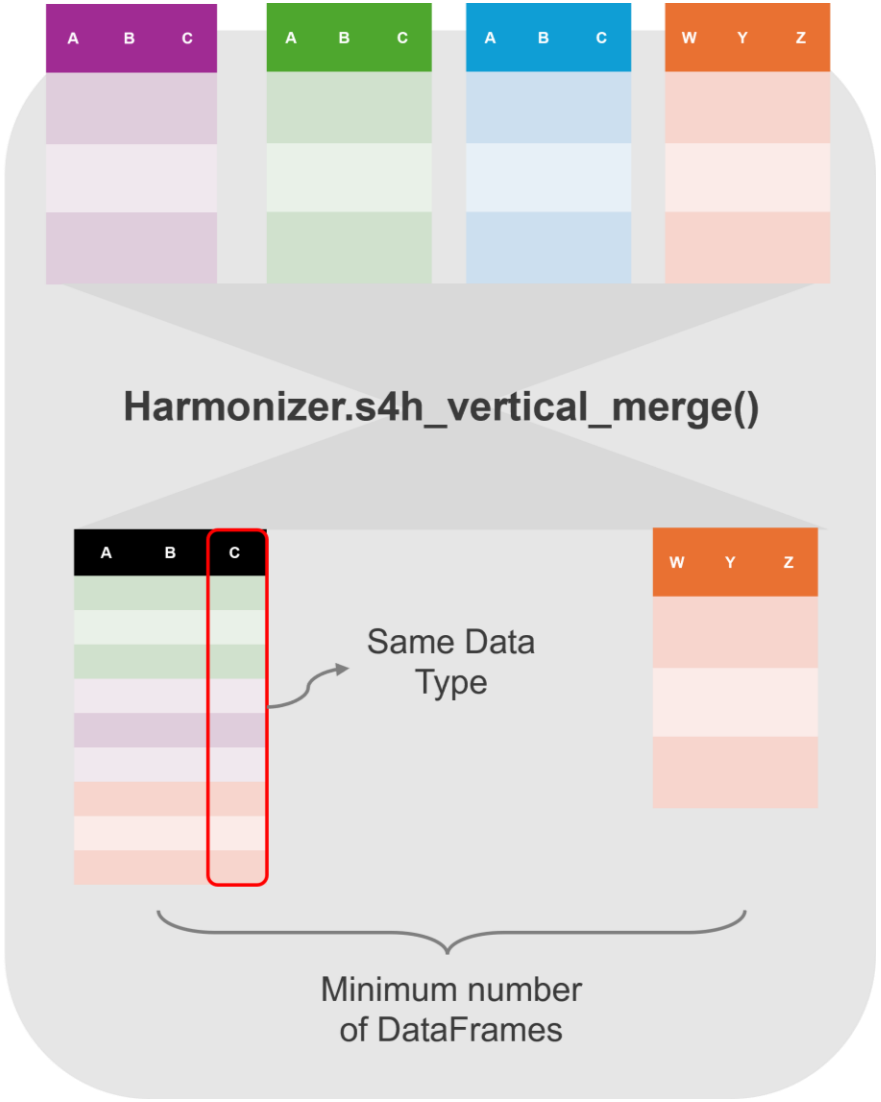
Standardized dictionary dataframe

Output

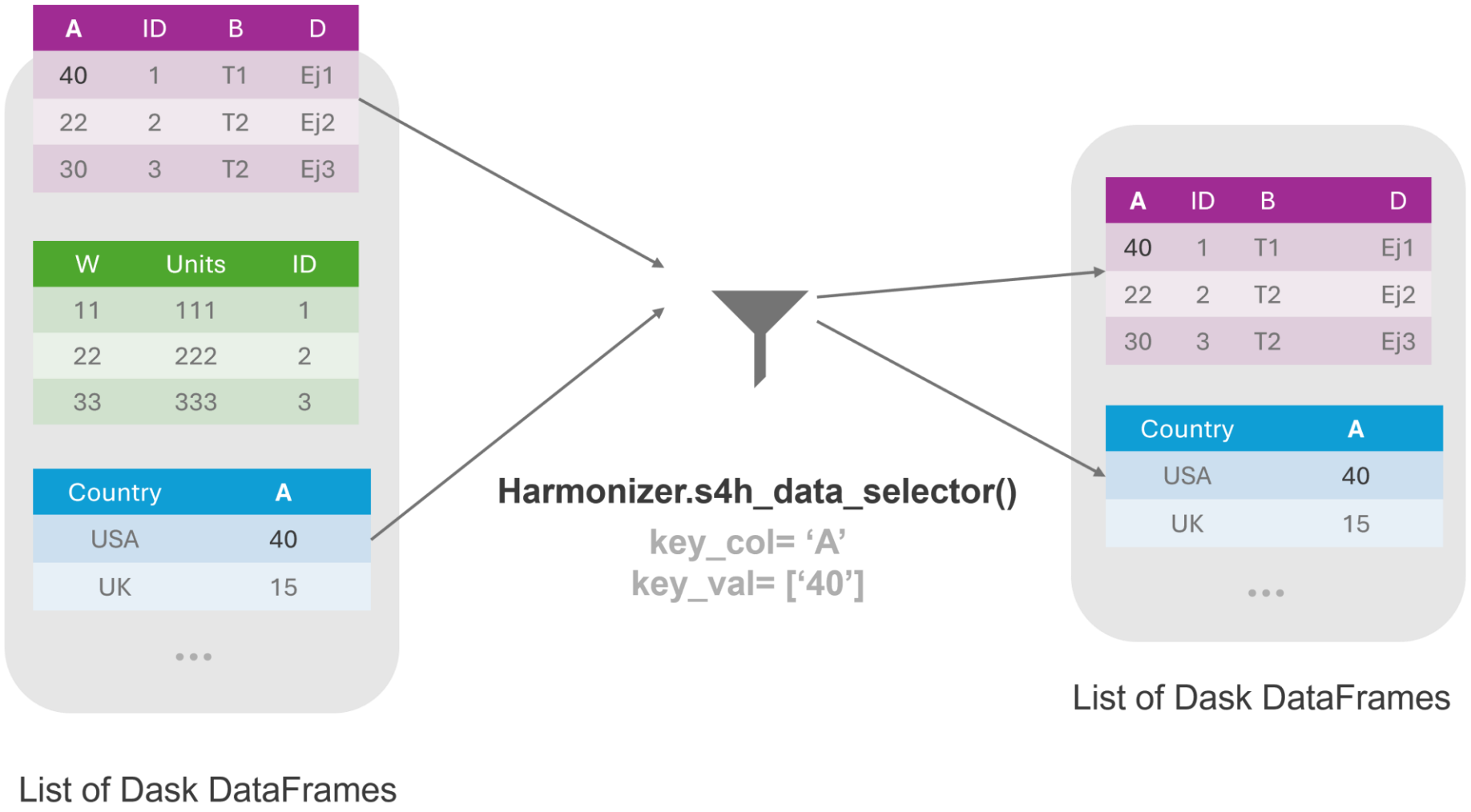
Unified pandas dataframe

Default categories: Business, Education, Fertility, Housing, Identification, Migration, Nonstandard job, Social Security

Harmonizer: Vertical Merge



Harmonizer: Data Selector



Harmonizer: Data Joining

List of Dask DataFrames

A	ID	B	D
40	1	T1	Ej1
22	2	T2	Ej2
30	3	T2	Ej3

W	Units	ID
11	111	1
22	222	2
33	333	3

Country	ID
USA	1
UK	2

...

Harmonizer.s4h.join_data()

ID	Country	Units	A	B	D	W
1	USA	NA	40	T1	Ej1	NA
1	USA	111	NA	NA	NA	11
2	UK	NA	22	T2	Ej2	NA
2	UK	222	NA	NA	NA	22
3	NA	NA	30	T2	Ej3	NA
3	NA	333	NA	T2	NA	33

Pandas DataFrame

Harmonizer utils: Dictionary

Standardize dictionary

Cleans and **structures** the dictionary by standardizing text fields, grouping answers, and removing duplicates.

Translate column

- **Column:** Name of the column to translate.
- **Language:** Target language code (“en”, “es”, “pt”).

Classify rows

- **Model path:** Location of the model weights.
- **Columns:** columns containing survey-related text
- **New column:** corresponding to the classified group

Input

Formatted Dictionary xlxs

Classification Model (BERT)

Google Translate API

Output

Standardized dictionary dataframe

Default categories: Business, Education, Fertility, Housing, Identification, Migration, Nonstandard job, Social Security

Demo

github.com/harmonize-tools/interfaz_s4h

Demo

Follow along with a demonstration of the library pipeline using Streamlit's GUI

What we will learn:

- How to work with data from an annual survey
- How to use user interface
- How to extract local data

INEI INSTITUTO NACIONAL DE ESTADISTICA E INFORMATICA

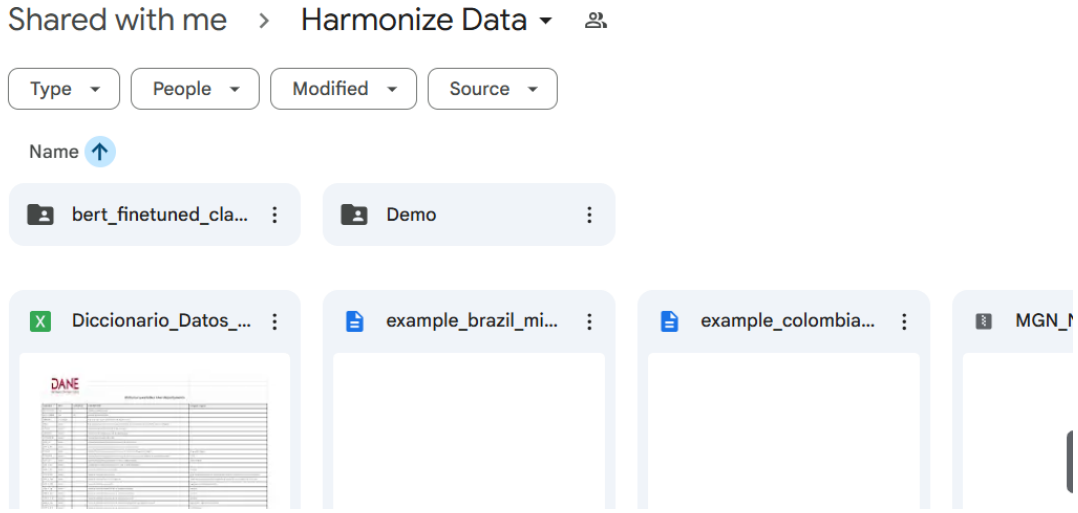
Peruvian National Housing Survey (**ENAHO 2022 and 2023**)

Monitoring of indicators on living conditions at the national level, in urban and rural areas, in the country's 24 departments

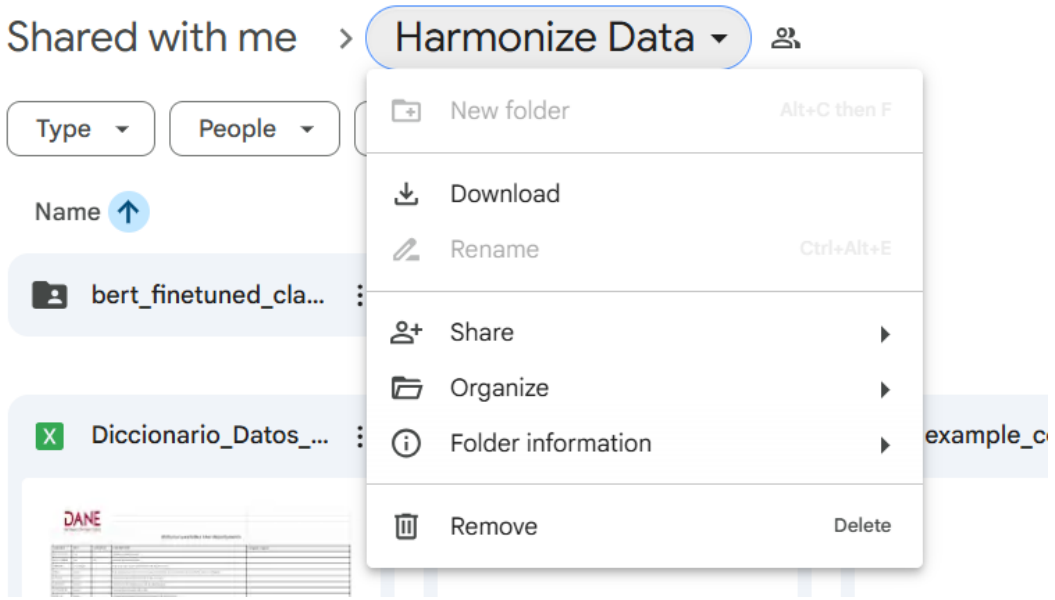
Session Data

bit.ly/3WZp0dG

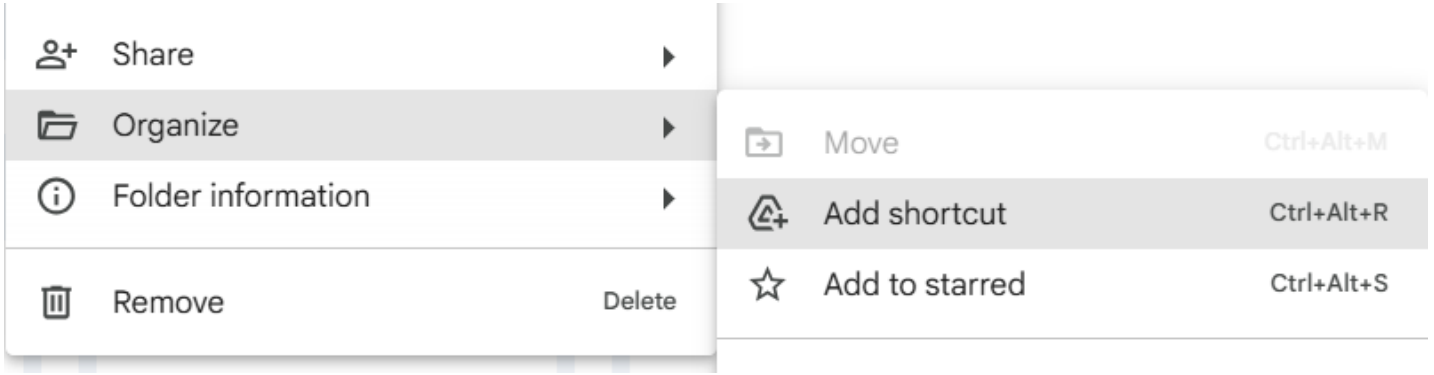
1


Shared with me
Harmonize Data
Type
People
Modified
Source
Name
bert_finetuned_cla...
Demo
Diccionario_Datos_...
example_brazil_mi...
example_colombia...
MGN_

2


Shared with me
Harmonize Data
Type
People
Name
bert_finetuned_cla...
Diccionario_Datos_...
New folder
Alt+C then F
Download
Rename
Ctrl+Alt+E
Share
Organize
Folder information
Remove
Delete

3


Share
Organize
Folder information
Remove
Delete
Move
Ctrl+Alt+M
Add shortcut
Ctrl+Alt+R
Add to starred
Ctrl+Alt+S

Dictionary

bit.ly/3WZp0dG

Data and Dictionary

Diccionario_Datos_Niveles_Variables_CNPV2018.xlsx

MGN_ANM_DPTOS | MGN_ANM_MPIOS | MGN_ANM_MPIOCL | MGN_ANM_SECTOR_RURAL | MGN_ANM_SECCION_RURAL

DANE
INFORMACIÓN PARA TODOS

Atributos y variables nivel municipio

VARIABLE	TIPO	LONGITUD	DESCRIPCIÓN	Categoria original
DPTO_CCDGO	Text	2	Código del departamento	
MPIO_CCDGO	Text	3	Código que identifica al municipio	
MPIO_CNMBR	Text	250	Nombre del municipio	
MPIO_CDPMP	Text	5	Código concatenado que identifica al municipio	
VERSION	Long Integer		Año de la información geográfica	
AREA	Double		Área del municipio en metros cuadrados (Sistema de coordenadas planas MAGNA_Colombia_Bogota)	
LATITUD	Double		Coordenada de latitud centroide del municipio	
LONGITUD	Double		Coordenada de longitud centroide del municipio	
STCTNENCUE	Double		Cantidad de Encuestas CNPV 2018	
STP3_1_SI	Double		Cantidad de encuestas que reportaron estar en territorio étnico	
STP3_2_NO	Double		Cantidad de encuestas que reportaron no estar en territorio étnico	
STP3A_RI	Double		Cantidad de encuestas que reportaron estar en territorio étnico Resguardo indígena	Resguardo Indígena
STP3B_TCN	Double		Cantidad de encuestas que reportaron estar en territorio étnico Territorio colectivo de comunidades negras	TCCN
STP4_1_SI	Double		Cantidad de encuestas que reportaron estar en áreas protegidas	Área protegida
STP4_2_NO	Double		Cantidad de encuestas que reportaron no estar en áreas protegidas	
STP9_1_USO	Double		Conteo de unidades con uso vivienda	Vivienda
STP9_2_USO	Double		Conteo de unidades con uso mixto	Mixto (Espacio independiente y separado que combina vivienda con otro uso no residencial)
STP9_3_USO	Double		Conteo de unidades con uso no residencial	Unidad NO Residencial (Espacio independiente y separado con uso diferente a vivienda)

Data and Dictionary

variable_name	type	size	question	description	value		

Data and Dictionary

template.xlsx

variable_name	type	size	question	description	value
DPTO_CCDGO	Text	2	Código del departamento		
MPIO_CCDGO	Text	3	Código que identifica al municipio		
MPIO_CNMBR	Text	250	Nombre del municipio		
MPIO_CDPMP	Text	5	Código concatenado que identifica al municipio		
VERSION	Long Integer		Año de la información geográfica		
AREA	Double		Área del municipio en metros cuadrados (Sistema de coordenadas planas MAGNA_Colombia_Bogota)		
LATITUD	Double		Coordenada de latitud centroide del municipio		
LONGITUD	Double		Coordenada de longitud centroide del municipio		
STCTNENCUE	Double		Cantidad de Encuestas CNPV 2018		
STP3_1_SI	Double		Cantidad de encuestas que reportaron estar en territorio étnico		
STP3_2_NO	Double		Cantidad de encuestas que reportaron no estar en territorio étnico		
STP3A_RI	Double		Cantidad de encuestas que reportaron estar en territorio étnico Resguardo indígena	Resguardo Indígena	
STP3B_TCN	Double		Cantidad de encuestas que reportaron estar en territorio étnico Territorio colectivo de comunidades negras	TCCN	
STP4_1_SI	Double		Cantidad de encuestas que reportaron estar en áreas protegidas	Área protegida	
STP4_2_NO	Double		Cantidad de encuestas que reportaron no estar en áreas protegidas		
STP9_1_USO	Double		Conteo de unidades con uso vivienda	Vivienda	
STP9_2_USO	Double		Conteo de unidades con uso mixto	Mixto (Espacio independiente y separado que combina vivienda	
STP9_3_USO	Double		Conteo de unidades con uso no residencial	Unidad NO Residencial (Espacio independiente y separado cor	
STP9_4_USO	Double		Conteo de unidades con uso LEA	Lugar especial de alojamiento (LEA)	
STP9_2_1_M	Double		Conteo de unidades mixtas con uso no residencial industria	Industria	
STP9_2_2_M	Double		Conteo de unidades mixtas con uso no residencial comercio	Comercio	
STP9_2_3_M	Double		Conteo de unidades mixtas con uso no residencial servicios	Servicios	
STP9_2_4_M	Double		Conteo de unidades mixtas con uso no residencial agropecuario, agroindustrial, forestal	Agropecuario, Agroindustrial, Forestal	
STP9_2_9_M	Double		Conteo de unidades mixtas con uso no residencial sin información	Sin información	
STP9_3_1_N	Double		Conteo de unidades no residénciales con uso Industria	Industria	
STP9_3_2_N	Double		Conteo de unidades no residénciales con uso Comercio	Comercio	

Hands-on

bit.ly/3WZp0dG

harmonize-tools.github.io/socio4health

Hands-on

Socioeconomic and demographic variables on dengue incidence in Colombia

In this example we will use **socio4health** to **retrieve, harmonize** and **analyze socioeconomic and demographic** variables related to **dengue** incidence in Colombia and recreate the dataset used in the publication

viruses

MDPI

Article

Exploring Dengue Dynamics: A Multi-Scale Analysis of Spatio-Temporal Trends in Ibagué, Colombia

Julian Otero [1,2,*], Alejandra Tabares [3,†] and Mauricio Santos-Vega [2,4,†]

DANE

Colombian National Population and Housing Census 2018 (**CNPV 2018**)

This tutorial assumes you have an **intermediate** or **advanced** understanding of **Python** and **data manipulation**

What we will learn:

- Local extraction data
- Shape file processing

socio4health

Install **socio4health** using the following command:

```
pip install socio4health
```

The package requires **Python version 3.10** or higher

Connect **Google Colab** notebook to your drive

```
google.colab import drive
drive.mount('/content/drive')
```

Hands-on

1. Import the following libraries

```
import datetime
import geopandas as gpd
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils
```

⚠️ **Make sure you have installed the package**

2. Extract data from Colombia National Population and Housing Census 2018

```
col_local_extractor = Extractor(
input_path="",
down_ext=[],
output_path="",
key_words=[])
col_CNPV = col_local_extractor.s4h_extract()
```

Hands-on

1. Import the following libraries

```
import datetime
import geopandas as gpd
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils
```

⚠️ **Make sure you have installed the package**

2. Extract data from Colombia National Population and Housing Census 2018

```
col_local_extractor = Extractor(
input_path="/content/drive/MyDrive/Harmonize Data/Example Colombia/",
down_ext=[],
output_path=,
key_words=[])
col_CNPV = col_local_extractor.s4h_extract()
```

Hands-on

1. Import the following libraries

```
import datetime
import geopandas as gpd
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils
```

⚠️ **Make sure you have installed the package**

2. Extract data from Colombia National Population and Housing Census 2018

```
col_local_extractor = Extractor(
input_path="/content/drive/MyDrive/Harmonize Data/Example Colombia/",
down_ext=['.cpg', '.dbf', '.prj', '.sbn', '.sbx', '.shx', '.shp', '.zip'],
output_path=,
key_words=[])
col_CNPV = col_local_extractor.s4h_extract()
```

Hands-on

1. Import the following libraries

```
import datetime
import geopandas as gpd
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils
```

⚠️ **Make sure you have installed the package**

2. Extract data from Colombia National Population and Housing Census 2018

```
col_local_extractor = Extractor(
input_path="/content/drive/MyDrive/Harmonize Data/Example Colombia/",
down_ext=['.cpg', '.dbf', '.prj', '.sbn', '.sbx', '.shx', '.shp', '.zip'],
output_path="CNVP2018",
key_words=[])
col_CNPV = col_local_extractor.s4h_extract()
```

Hands-on

1. Import the following libraries

```
import datetime
import geopandas as gpd
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils
```

⚠️ **Make sure you have installed the package**

2. Extract data from Colombia National Population and Housing Census 2018

```
col_local_extractor = Extractor(
input_path="/content/drive/MyDrive/Harmonize Data/Example Colombia/",
down_ext=['.cpg', '.dbf', '.prj', '.sbn', '.sbx', '.shx', '.shp', '.zip'],
output_path="CNVP2018",
key_words=['MGN_NivelMunicipioIntegrado_CNPV.zip'])
col_CNPV = col_local_extractor.s4h_extract()
```

Hands-on

```
1 col_CNPV[0]
```

	DPTO_CCDGO	MPIO_CCDGO	MPIO_CNMBR	MPIO_CDPMP	VERSION	AREA	LATITUD	LONGITUD	STCTNENCUE	STP3_1_SI	...	STP51_PRIM	STP51_SECU	STP51_SUPE	STP51_POST
0	18	001	FLORENCIA	18001	2018	2.547638e+09	1.749139	-75.558239	71877.0	32.0	...	37918.0	14123.0	14606.0	856.0
1	18	029	ALBANIA	18029	2018	4.141221e+08	1.227865	-75.882327	2825.0	24.0	...	1696.0	150.0	98.0	0.0
2	18	094	BELÉN DE LOS ANDAQUÍES	18094	2018	1.191619e+09	1.500923	-75.875645	4243.0	54.0	...	2596.0	418.0	171.0	12.0
3	18	247	EL DONCELLO	18247	2018	1.106076e+09	1.791386	-75.193944	8809.0	0.0	...	6091.0	712.0	347.0	26.0
4	18	256	EL PAUJÍL	18256	2018	1.234734e+09	1.617746	-75.234043	5795.0	0.0	...	4805.0	261.0	226.0	0.0

Hands-on

3. Load the dictionary

```
raw_dic = pd.read_excel("")
```

4. Standardize the dictionary and translate the question, description and possible_answers columns

```
dic = harmonizer_utils.s4h_standardize_dict(raw_dict=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
```

5. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = harmonizer_utils.s4h_classify_rows()
```

Hands-on

3. Load the dictionary

```
raw_dic = pd.read_excel("/content/drive/MyDrive/Harmonize Data/Example Colombia/raw_dic_mpio.xlsx")
```

4. Standardize the dictionary and translate the question, description and possible_answers columns

```
dic = harmonizer_utils.s4h_standardize_dict(raw_dict=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
```

5. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = harmonizer_utils.s4h_classify_rows()
```

Hands-on

3. Load the dictionary

```
raw_dic = pd.read_excel("/content/drive/MyDrive/Harmonize Data/Example Colombia/raw_dic_mpio.xlsx")
```

4. Standardize the dictionary and translate the question, description and possible_answers columns

```
dic = harmonizer_utils.s4h_standardize_dict(raw_dict=raw_dic)
dic = harmonizer_utils.s4h_translate_column(data=dic, column="question",
                                            language="en")
dic = harmonizer_utils.s4h_translate_column(data=dic, column="description",
                                            language="en")
dic = harmonizer_utils.s4h_translate_column(data=dic, column="possible_answers",
                                            language="en")
```

5. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = harmonizer_utils.s4h_classify_rows(data=, col1=, col2=, col3=,
new_column_name=, MODEL_PATH=)
```

Hands-on

3. Load the dictionary

```
raw_dic = pd.read_excel("/content/drive/MyDrive/Harmonize Data/Example Colombia/raw_dic_mpio.xlsx")
```

4. Standardize the dictionary and translate the question, description and possible_answers columns

```
dic = harmonizer_utils.s4h_standardize_dict(raw_dict=raw_dic)
dic = harmonizer_utils.s4h_translate_column(data=dic, column="question",
                                            language="en")
dic = harmonizer_utils.s4h_translate_column(data=dic, column="description",
                                            language="en")
dic = harmonizer_utils.s4h_translate_column(data=dic, column="possible_answers",
                                            language="en")
```

5. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = harmonizer_utils.s4h_classify_rows(data=dic,
col1="question_en", col2="description_en", col3="possible_answers_en",
new_column_name="category",
MODEL_PATH="/content/drive/MyDrive/Harmonize Data/bert_finetuned_classifier")
```

```
dic
```

	variable_name	type	size	question	description	value	possible_answers	question_en	description_en	possible_answers_en	category
0	VERSION	Long Integer	NaN	año de la información geográfica	NaN	NaN	NaN	year of geographic information	NaN	NaN	Identification
1	STCTNENCUE	Double	NaN	cantidad de encuestas cnpv 2018	NaN	NaN	NaN	number of cnpv surveys 2018	NaN	NaN	Identification
2	STP3_1_SI	Double	NaN	cantidad de encuestas que reportaron estar en ...	NaN	NaN	NaN	number of surveys that reported being in ethni...	NaN	NaN	Identification
3	STP3A_RI	Double	NaN	cantidad de encuestas que reportaron estar en ...	resguardo indígena	NaN	NaN	number of surveys that reported being in indig...	indigenous reservation	NaN	Identification
4	STP3B_TCN	Double	NaN	cantidad de encuestas que reportaron estar en ...	tccn	NaN	NaN	number of surveys that reported being in ethni...	tccn	NaN	Identification
...	...	...	...	...	...	...	...	...	...	...	...
83	DPTO_CCDGO	Text	2.0	código del departamento	NaN	NaN	NaN	department code	NaN	NaN	Business
84	MPIO_CCDGO	Text	3.0	código que identifica al municipio	NaN	NaN	NaN	code that identifies the municipality	NaN	NaN	Identification
85	MPIO_CNMBR	Text	250.0	nombre del municipio	NaN	NaN	NaN	name of the municipality	NaN	NaN	Identification
86	STP27_PERS	Double	NaN	número de personas	NaN	NaN	NaN	number of people	NaN	NaN	Identification
87	AREA	Double	NaN	área del municipio en metros cuadrados (sistem...	NaN	NaN	NaN	area of the municipality in square meters (m...	NaN	NaN	Business

88 rows × 11 columns

Hands-on

6. Create an instance of Harmonizer class

```
har = Harmonizer()
```

7. Set the similarity threshold to 0.9 and NaN threshold to 1 for data selection

```
har.similarity_threshold =
har.nan_threshold =
```

8. Display available columns in the DataFrame

```
available_columns =  har.s4h_get_available_columns()
available_columns
```

Hands-on

6. Create an instance of Harmonizer class

```
har = Harmonizer()
```

7. Set the similarity threshold to 0.9 and NaN threshold to 1 for data selection

```
har.similarity_threshold = 0.9
har.nan_threshold = 1
```

8. Display available columns in the DataFrame

```
available_columns =  har.s4h_get_available_columns()
available_columns
```

[ ]

```
har = Harmonizer()
har.similarity_threshold = 0.9
har.nan_threshold = 1
available_columns = har.s4h_get_available_columns(col_CNPV)
available_columns
```

```
['AREA',
 'DPTO_CCDGO',
 'LATITUD',
 'LONGITUD',
 'MPIO_CCDGO',
 'MPIO_CDPMP',
 'MPIO_CNMBR',
 'STCTNENCUE',
 'STP14_1_TI',
 'STP14_2_TI',
 'STP14_3_TI',
 'STP14_4_TI',
 'STP14_5_TI',
 'STP14_6_TI',
 'STP15_1_OC',
 'STP15_2_OC',
 'STP15_3_OC',
 'STP15_4_OC',
 'STP19_ACU1',
 'STP19_ACU2',
 'STP19_ALC1',
 'STP19_ALC2',
 'STP19_EC_1',
 'STP19_EE_1',
 'STP19_EE_2',
 'STP19_EE_3',
 'STP19_EE_4',
 'STP19_EE_5',
 'STP19_EE_6',
 'STP19_EE_9',
 'STP19_ES_2',
 'STP19_GAS1',
```

Hands-on

9. Set other parameters for data selection

```
har.dict_df = dic
har.categories = [""]
har.extra_cols = ['']
har.key_col = ''
```

10. Create a subset of the data

```
filtered_ddfs = har.s4h_data_selector()
```

Hands-on

9. Set other parameters for data selection

```
har.dict_df = dic
har.categories = ["Housing"]
har.extra_cols = ['MPIO_CDPMP', 'GEOMETRY']
har.key_col = 'MPIO_CDPMP'
```

10. Create a subset of the data

```
filtered_ddfs = har.s4h_data_selector()
```

Hands-on

9. Set other parameters for data selection

```
har.dict_df = dic
har.categories = ["Housing"]
har.extra_cols = ['MPIO_CDPMP', 'GEOMETRY']
har.key_col = 'MPIO_CDPMP'
```

10. Create a subset of the data

```
filtered_ddfs = har.s4h_data_selector(col_CNPV)
```

```
1 filtered_ddfs[0]
```

	MPIO_CDPMP	STPERSON_S	STP9_4_USO	STP9_2_USO	STP9_3_USO	STP9_1_USO	STP9_2_3_M	STP9_2_9_M	STP9_3_4_N	STP9_3_10	...	STP19_INT2	STP19_REC2	STP14_2_TI	STP14_1_T
0	18001	152474.0	87.0	2178.0	8436.0	61176.0	566.0	5.0	535.0	371.0	...	35727.0	4318.0	13764.0	47817
1	18029	4363.0	2.0	49.0	948.0	1826.0	12.0	0.0	728.0	14.0	...	1370.0	682.0	40.0	1793
2	18094	8729.0	11.0	109.0	900.0	3223.0	6.0	0.0	2.0	93.0	...	2775.0	978.0	113.0	3189
3	18247	17572.0	4.0	357.0	1850.0	6598.0	87.0	0.0	807.0	39.0	...	5395.0	1419.0	775.0	6006
4	18256	12822.0	5.0	204.0	695.0	4891.0	38.0	0.0	4.0	45.0	...	4379.0	2154.0	145.0	4700

Hands-on

11. Compare the available columns in the dataset with the variables in the dictionary

```
har.s4h_compare_with_dict()
```

12. Use the filtered DataFrame to explore the harmonized data of specified municipality. You can either export it as a CSV file or visualize it by **matplotlib**, **geopandas** and **numpy**

Hands-on

11. Compare the available columns in the dataset with the variables in the dictionary

```
har.s4h_compare_with_dict(col_CNPV)
```

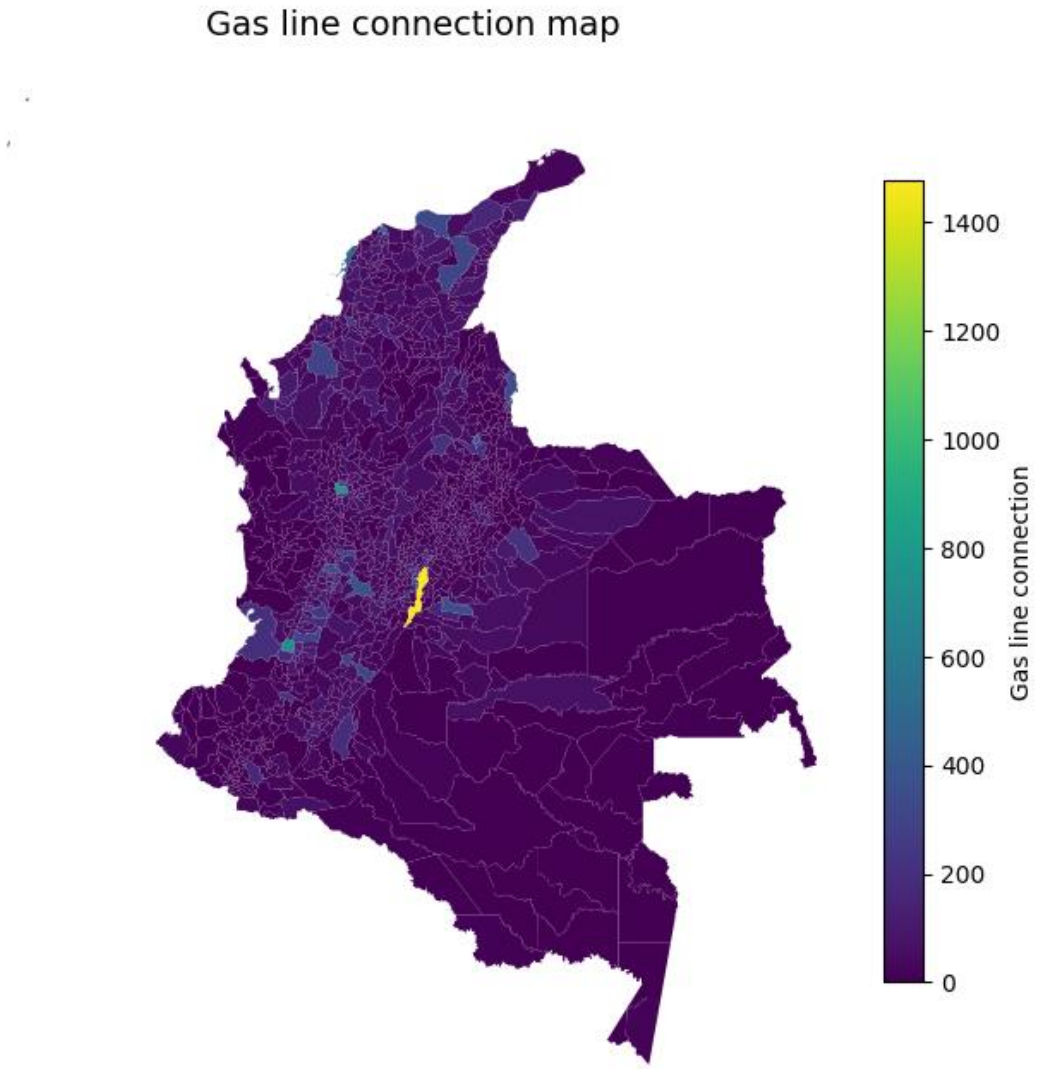
12. Use the filtered DataFrame to explore the harmonized data of specified municipality. You can either export it as a CSV file or visualize it by **matplotlib**, **geopandas** and **numpy**

Hands-on

```
har.s4h_compare_with_dict(col_CNPV)
```

```
Matches with dict_df: 88 (95.65%)
```

	Unmatched ddfs variable	Unmatched dict_df variables
0	FILENAME	None
1	GEOMETRY	None
2	SHAPE_AREA	None
3	SHAPE_LENG	None



Hands-on

Socioeconomic and demographic variables on dengue incidence in Brazil

In this example we will use **socio4health** to **retrieve, harmonize** and **analyze socioeconomic and demographic**, such as the level of urbanization and access to water supply in Brazil, to recreate the dataset used in the publication

Combined effects of hydrometeorological hazards and urbanisation on dengue risk in Brazil: a spatiotemporal modelling study

Rachel Lowe, Sophie A Lee, Kathleen M O'Reilly, Oliver J Brady, Leonardo Bastos, Gabriel Carrasco-Escobar, Rafael de Castro Catão, Felipe J Colón-González, Christovam Barcellos, Marilia Sá Carvalho, Marta Blangiardo, Håvard Rue, Antonio Gasparrini

IBGE
Instituto Brasileiro de Geografia e Estatística

Brazilian Demographic census 2010

This tutorial assumes you have an **intermediate** or **advanced** understanding of **Python** and **data manipulation**

What we will learn:

- Online extraction data (web scraping)
- Fixed-width format file processing

Hands-on

1. Import the following libraries

```
import re
import pandas as pd
import dask.dataframe as dd
import matplotlib.pyplot as plt
from matplotlib.ticker import FuncFormatter
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils, extractor_utils
```

2. Load the dictionary

```
raw_dic = pd.read_excel("")
```

3. Standardize the dictionary

```
dic = harmonizer_utils.s4h_standardize_dict()
```

⚠️ **Make sure you have installed the package**

Hands-on

1. Import the following libraries

```
import re
import pandas as pd
import dask.dataframe as dd
import matplotlib.pyplot as plt
from matplotlib.ticker import FuncFormatter
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils, extractor_utils
```

2. Load the dictionary

```
raw_dic = pd.read_excel("/content/drive/MyDrive/Harmonize
Data/Example Brazil/raw_dictionary_br_2010.xlsx")
```

3. Standardize the dictionary

```
dic = harmonizer_utils.s4h_standardize_dict()
```

⚠️ **Make sure you have installed the package**

Hands-on

1. Import the following libraries

```
import re
import pandas as pd
import dask.dataframe as dd
import matplotlib.pyplot as plt
from matplotlib.ticker import FuncFormatter
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils, extractor_utils
```

2. Load the dictionary

```
raw_dic = pd.read_excel("/content/drive/MyDrive/Harmonize
Data/Example Brazil/raw_dictionary_br_2010.xlsx")
```

3. Standardize the dictionary

```
dic = harmonizer_utils.s4h_standardize_dict(raw_dic)
```

⚠️ **Make sure you have installed the package**

Hands-on

4. Since the format used by **IBGE** is **FWF** (fixed-width file), use the utility functions provided to complete the standardization process

```
colnames, colspecs = extractor_utils.s4h_parse_fwf_dict()
```

5. Translate columns in the dictionary

```
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
```

6. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = dic = harmonizer_utils.s4h_classify_rows(data=, col1=, col2=, col3=,
new_column_name=, MODEL_PATH=)
```

Hands-on

4. Since the format used by **IBGE** is **FWF** (fixed-width file), use the utility functions provided to complete the standardization process

```
colnames, colspecs =extractor_utils.s4h_parse_fwf_dict(dic)
```

5. Translate columns in the dictionary

```
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
dic = harmonizer_utils.s4h_translate_column(data=, column=, language=)
```

6. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = harmonizer_utils.s4h_classify_rows(data=, col1=, col2=, col3=,
new_column_name=, MODEL_PATH=)
```

Hands-on

4. Since the format used by **IBGE** is **FWF** (fixed-width file), use the utility functions provided to complete the standardization process

```
colnames, colspecs =extractor_utils.s4h_parse_fwf_dict(dic)
```

5. Translate columns in the dictionary

```
dic = harmonizer_utils.s4h_translate_column(dic, "question", language="en")
dic = harmonizer_utils.s4h_translate_column(dic, "description", language="en")
dic = harmonizer_utils.s4h_translate_column(dic, "possible_answers",
language="en")
```

6. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = harmonizer_utils.s4h_classify_rows(data=, col1=, col2=, col3=,
new_column_name=, MODEL_PATH=)
```

Hands-on

4. Since the format used by **IBGE** is **FWF** (fixed-width file), use the utility functions provided to complete the standardization process

```
colnames, colspecs =extractor_utils.s4h_parse_fwf_dict(dic)
```

5. Translate columns in the dictionary

```
dic = harmonizer_utils.s4h_translate_column(dic, "question", language="en")
dic = harmonizer_utils.s4h_translate_column(dic, "description", language="en")
dic = harmonizer_utils.s4h_translate_column(dic, "possible_answers",
language="en")
```

6. Classify the dictionary rows using the pre-trained **BERT model** 

```
dic = harmonizer_utils.s4h_classify_rows(data=dic,
col1="question_en", col2="description_en", col3="possible_answers_en",
new_column_name="category",
MODEL_PATH="/content/drive/MyDrive/Harmonize Data/bert_finetuned_classifier")
```

```
7 dic
```

```
question translated
description translated
possible_answers translated
Device set to use cpu
```

	variable_name	question	description	value	initial_position	final_position	size	dec	type	possible_answers	question_en	description_en	possible_answers_en
0	V0402	a responsabilidade pelo domicílio é de:	NaN	1.0; 2.0; 9.0	107.0	107.0	1.0	NaN	C	apenas um morador; mais de um morador; ignorado	Responsibility for the home is:	NaN	just one resident; more than one resident; ign...
1	V0209	abastecimento de água, canalização:	NaN	1.0; 2.0; 3.0	90.0	90.0	1.0	NaN	C	sim, em pelo menos um cômodo; sim, só na propr...	water supply, plumbing:	NaN	yes, in at least one room; yes, only on the pr...
2	V0208	abastecimento de água, forma:	NaN	1.0; 2.0; 3.0; 4.0; 5.0; 6.0; 7.0; 8.0; 9.0; 10.0	88.0	89.0	2.0	NaN	C	rede geral de distribuição; poço ou nascente n...	water supply, form:	NaN	general distribution network; well or spring o...
3	V6210	adequação da moradia	NaN	1.0; 2.0; 3.0	144.0	144.0	1.0	NaN	C	adequada; semi-adequada; inadequada	suitability of housing	NaN	adequate; semi-adequate; inappropriate

Hands-on

7. Extract the Brazil Census 2010 dataset from the Brazilian Institute of Geography and Statistics (IBGE)

```
bra_online_extractor = Extractor(
input_path=,
down_ext=[], output_path=, key_words=[],depth=, is_fwf=, colnames=,
colspecs=)
bra_Censo_2010 = bra_online_extractor.s4h_extract()
```

8. Create a Harmonizer class instance and set the similarity threshold to 0.9

```
har = Harmonizer()
har.similarity_threshold =
```

9. Merge DataFrames vertically

```
dfs = har.s4h_vertical_merge()
```

Hands-on

7. Extract the Brazil Census 2010 dataset from the Brazilian Institute of Geography and Statistics (IBGE)

```
bra_online_extractor = Extractor(
input_path="https://www.ibge.gov.br/estatisticas/sociais/saude/9662-censo-demografico-2010.html?=&t=microdados",
down_ext=['.txt','.zip'], output_path="IBGE_2010", key_words=["^RJ\.zip$"],
depth=0, is_fwf=True, colnames=colnames, colspecs=colspecs)
bra_Censo_2010 = bra_online_extractor.s4h_extract()
```

8. Create a Harmonizer class instance and set the similarity threshold to 0.9

```
har = Harmonizer()
har.similarity_threshold =
```

9. Merge DataFrames vertically

```
dfs = har.s4h_vertical_merge()
```

Hands-on

7. Extract the Brazil Census 2010 dataset from the Brazilian Institute of Geography and Statistics (IBGE)

```
bra_online_extractor = Extractor(
input_path="https://www.ibge.gov.br/estatisticas/sociais/saude/9662-censo-demografico-2010.html?=&t=microdados",
down_ext=['.txt','.zip'], output_path="IBGE_2010", key_words=["^RJ\.zip$"],
depth=0, is_fwf=True, colnames=colnames, colspecs=colspecs)
bra_Censo_2010 = bra_online_extractor.s4h_extract()
```

8. Create a Harmonizer class instance and set the similarity threshold to 0.9

```
har = Harmonizer()
har.similarity_threshold = 0.9
```

9. Merge DataFrames vertically

```
dfs = har.s4h_vertical_merge()
```

Hands-on

7. Extract the Brazil Census 2010 dataset from the Brazilian Institute of Geography and Statistics (IBGE)

```
bra_online_extractor = Extractor(
input_path="https://www.ibge.gov.br/estatisticas/sociais/saude/9662-censo-demografico-2010.html?=&t=microdados",
down_ext=['.txt','.zip'], output_path="IBGE_2010", key_words=["^RJ\.zip$"],
depth=0, is_fwf=True, colnames=colnames, colspecs=colspecs)
bra_Censo_2010 = bra_online_extractor.s4h_extract()
```

8. Create a Harmonizer class instance and set the similarity threshold to 0.9

```
har = Harmonizer()
har.similarity_threshold = 0.9
```

9. Merge DataFrames vertically

```
dfs = har.s4h_vertical_merge(bra_Censo_2010)
```

+ Code + Text

[ ]

```
bra_online_extractor = Extractor(input_path="https://www.ibge.gov.br/estatisticas/sociais/saude/9662-censo-demografico-2010.html?=&t=microdados",
                                 down_ext=['.txt','.zip'],
                                 output_path="IBGE_2010_",
                                 key_words=["^RJ\.zip$"],
                                 depth=0, is_fwf=True, colnames=colnames, colspecs=colspecs)
bra_Censo_2010 = bra_online_extractor.s4h_extract()
```

```
<>:4: SyntaxWarning: invalid escape sequence '\.'
<>:4: SyntaxWarning: invalid escape sequence '\.'
/tmp/ipython-input-2582708218.py:4: SyntaxWarning: invalid escape sequence '\.'
  key_words=["^RJ\.zip$"],
/usr/local/lib/python3.12/dist-packages/scrapy/utils/request.py:120: ScrapyDeprecationWarning: 'REQUEST_FINGERPRINTER_IMPLEMENTATION' is a deprecated setting.
It will be removed in a future version of Scrapy.
  return cls(crawler)
DEBUG:asyncio:Using selector: EpollSelector
INFO:standard:Successfully saved links to Output_scrap.json.
Downloading files: 100%|██████████| 1/1 [00:11<00:00, 11.91s/it]
Processing files: 100%|██████████| 3/3 [01:28<00:00, 29.34s/it]
```

```
har = Harmonizer()
har.similarity_threshold = 0.9
dfs = har.s4h_vertical_merge(bra_Censo_2010)
```

```
Grouping DataFrames: 100%|██████████| 3/3 [00:00<00:00, 112.80it/s]
Merging groups: 100%|██████████| 1/1 [00:00<00:00, 16.33it/s]
```

Hands-on

9. Set other parameters for data selection

```
har.dict_df =
har.categories = []
har.key_col =
```

10. Create a subset of the data

```
filtered_ddfs = har.s4h_data_selector()
```

12. Use the filtered DataFrame to explore the harmonized data. You can either export it as a CSV file or visualize it by **matplotlib**, **geopandas** and **numpy**, etc.

Hands-on

9. Set other parameters for data selection

```
har.dict_df = dic
har.categories = ["Business"]
har.key_col = 'V0002'
```

10. Create a subset of the data

```
filtered_ddfs = har.s4h_data_selector()
```

12. Use the filtered DataFrame to explore the harmonized data. You can either export it as a CSV file or visualize it by **matplotlib**, **geopandas** and **numpy**, etc.

Hands-on

9. Set other parameters for data selection

```
har.dict_df = dic
har.categories = ["Business"]
har.key_col = 'V0002'
```

10. Create a subset of the data

```
filtered_ddfs = har.s4h_data_selector(col_CNPV)
```

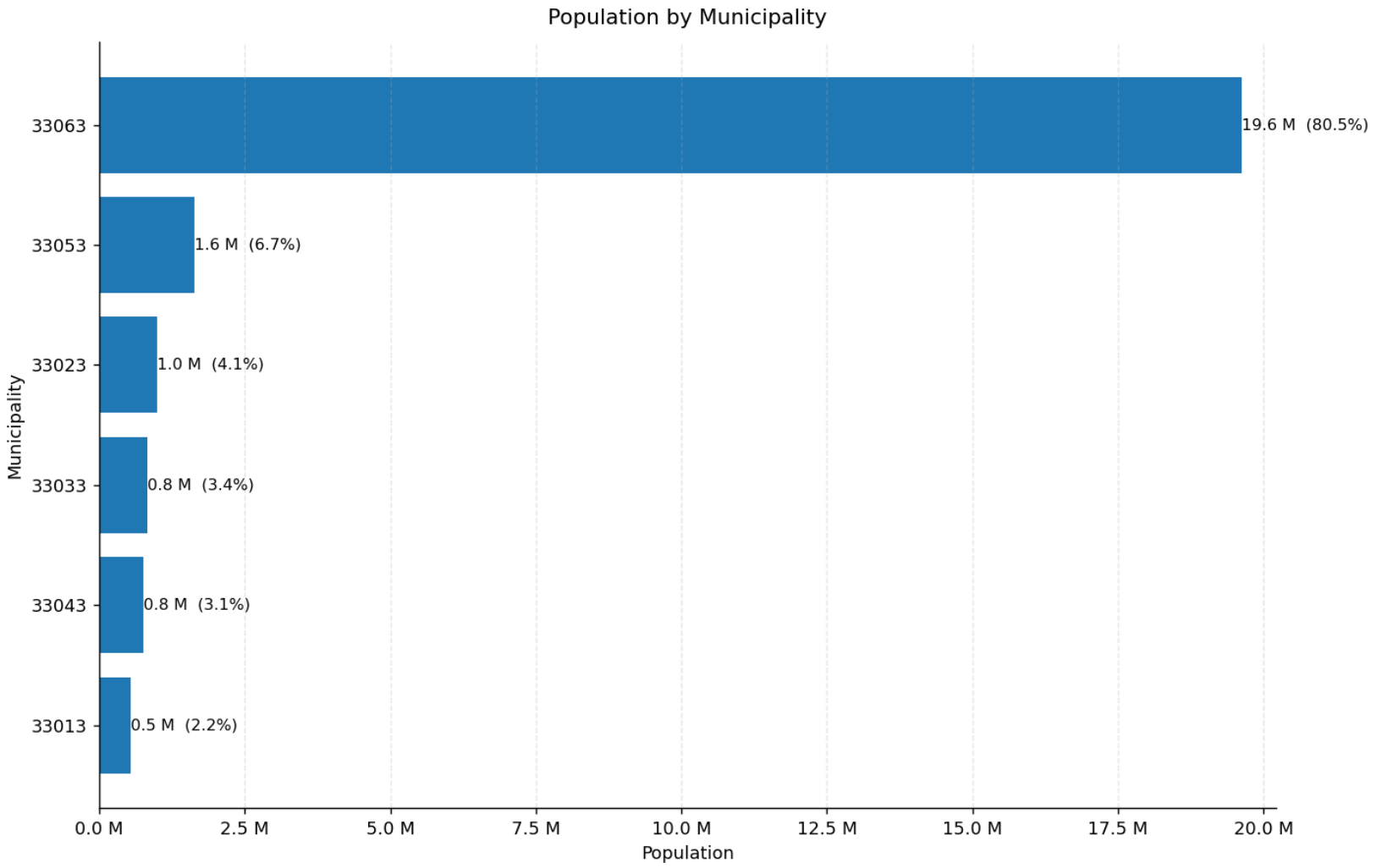
12. Use the filtered DataFrame to explore the harmonized data. You can either export it as a CSV file or visualize it by **matplotlib**, **geopandas** and **numpy**, etc.

```
1 filtered_ddfs[0].compute()
```

	V0002	V0208	V0301	V2012	V0222	V0701	V0211	V0207	V0212	M0201	...	V0221	V0401	V6531	V6532	V6530	V6529	V0206	V1005	V0001	V2011
0	33053	01	0	110 10010	1	<NA>	0	1	2	0	...	0	<NA>	0 0	0 001	0 0 0	0 0 0	1	1	33	010033
1	33053	01	0	110 20020	1	<NA>	0	1	2	0	...	0	<NA>	0 0	0 002	0 0 0	0 0 0	1	1	33	010033
2	33053	01	0	110120030	1	<NA>	0	1	2	0	...	0	<NA>	0 0	0 001	0 0 0	0 0 0	1	5	33	010033
3	33053	01	0	110 20010	2	2	0	1	2	0	...	0	40	010	0 001	383670540	0203019	1	2	33	010033
4	33053	01	0	110 10030	1	<NA>	0	1	2	0	...	0	<NA>	0 0	0 001	0 0 0	0 0 0	1	2	33	010033
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
442971	33053	02	0	210020210	2	0	0	1	1	0	...	0	20	004	<NA>	0000400602	2020200	0	<NA>	33	999004
442972	33053	01	0	210010010	2	0	0	1	1	0	...	0	20	001	<NA>	0000100201	2020100	0	<NA>	33	999004
442973	33053	01	0	210020110	2	0	0	1	1	0	...	0	10	503	<NA>	0000300601	2020200	0	<NA>	33	999004
442974	33053	01	0	210060210	2	0	0	1	5	0	...	0	10	708	<NA>	0000801302	2020100	0	<NA>	33	999004
442975	33053	01	0	210040810	2	0	0	1	4	0	...	0	20	0060303	<NA>	0001202403	2020100	0	<NA>	33	999004

2427048 rows × 47 columns

Hands-on


Population by Municipality
33063
33053
33023
33033
33043
33013
19.6 M (80.5%)
1.6 M (6.7%)
1.0 M (4.1%)
0.8 M (3.4%)
0.8 M (3.1%)
0.5 M (2.2%)
Municipality
0.0 M
2.5 M
5.0 M
7.5 M
10.0 M
12.5 M
15.0 M
17.5 M
20.0 M
Population

Next steps

Improve standarize dictionary process

Customize classifier and categories

Integrate different countries datastes

Include new countries

Feedback

How can socio4health improve?

Today you have used the 1.0.0 version of socio4health.

To keep improving and making the tool more useful to a bigger audience, we require the feedback from users.

Please fill in this form as detailed as possible
https://tinyurl.com/HARMONIZE-feedback

Thank you!

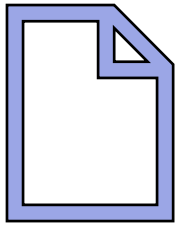
socio4health

HARMONIZE

Thank you!

FIOCRUZ, Rio de Janeiro - November 2025

uniandes FIOCRUZ IAI Inter-American Institute for Global Change Research BSC INPE UNIVERSIDAD PERUANA CAYETANO HEREDIA INDOMET INSTITUTO DOMINICANO DE METEOROLOGÍA wellcome



Extractor

Methods

`s4h_extract()`: Extract locally or online, files from a web page or zip file. Returns a list of dataframes.

`s4h_get_default_data_dir()`: Returns the default data directory for storing downloaded files.

`s4h_delete_download_folder()`: Safely delete the download folder and all its contents.

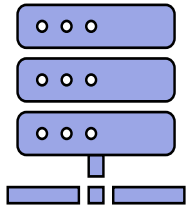
Utils

`s4h_compressed2files()`: Extract files from a compressed archive and return the paths of the extracted files.

`s4h_download_request()`: Download a file from the specified URL and save it to the given directory.

`s4h_run_standard_spider()`: Run the Scrapy spider to extract data from the given URL .

`s4h_parse_fwf_dict()`: Parse a dictionary DataFrame to extract column names and fixed-width format specifications



Harmonizer

Methods

`s4h_vertical_merge()`: Merge a list of Dask DataFrames vertically using instance parameters.

`s4h_drop_nan_columns()`: Drop columns where most values are `NaN` using instance parameters.

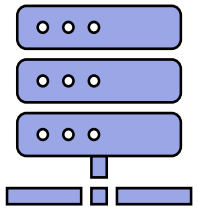
`s4h_get_available_columns()`: Get a list of unique column names from a single DataFrame or a list of DataFrames.

`s4h_harmonize_dataframes()`: Harmonize Dask DataFrames using the instance parameters.

`s4h_data_selector()`: Select rows from Dask DataFrames based on the instance parameters.

`s4h_compare_with_dict()`: Compare the columns available in the DataFrames with the variables in the dictionary and return a DataFrame with the columns that do not match in both directions.

`s4h_join_data()`: Join multiple Dask DataFrames on a specified `key_col`, removing duplicate columns.



Harmonizer

Utils

`s4h_classify_rows()`: Classify each row using a fine-tuned multiclass classification BERT model.

`s4h_get_classifier()`: Load the BERT fine-tuned model for classification only once.

`s4h_standardize_dict()`: Cleans and structures a dictionary-like DataFrame of variables by standardizing text fields, grouping possible answers, and removing duplicates.

`s4h_translate_column()`: Translates the content of selected columns in a DataFrame using Google Translate.

socio4health

How to use

1. Install **socio4health** using the following command:

```
pip install socio4health
```

The package requires **Python version 3.10** or higher

2. Import libraries

```
from socio4health import Extractor
from socio4health.harmonizer import Harmonizer
from socio4health.utils import harmonizer_utils
```

3. Create an instance of the Extractor and Harmonizer class:

```
extractor = Extractor(input_path='path/to/input',
down_ext=['.CSV'],sep=',', output_path='path/to/output')
harmonizer = Harmonizer()
```

HARMONIZE

socio4health

Graphic User Interface Tutorial

FIOCRUZ, Rio de Janeiro - November 2025

uniandes FIOCRUZ IAI Inter-American Institute for Global Change Research BSC INPE UNIVERSIDAD PERUANA CAYETANO HEREDIA INDOMET INSTITUTO DOMINICANO DE METEOROLOGÍA wellcome

How to use

To use the **socio4health** GUI fork or clone the interface from the **Github** repository:

https://github.com/harmonize-tools/interfaz_s4h.git

1. Create and activate a virtual environment (recommended):

```
python -m venv venv
```

Activate on **Windows**

```
venv\Scripts\activate
```

Activate on **macOS/Linux**

```
source venv/bin/activate
```

How to use

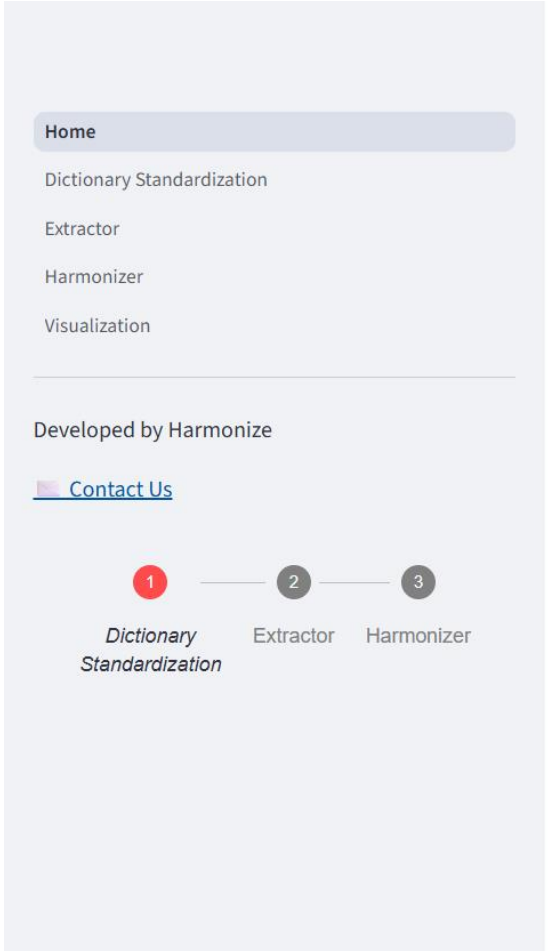
3. Install requirements:

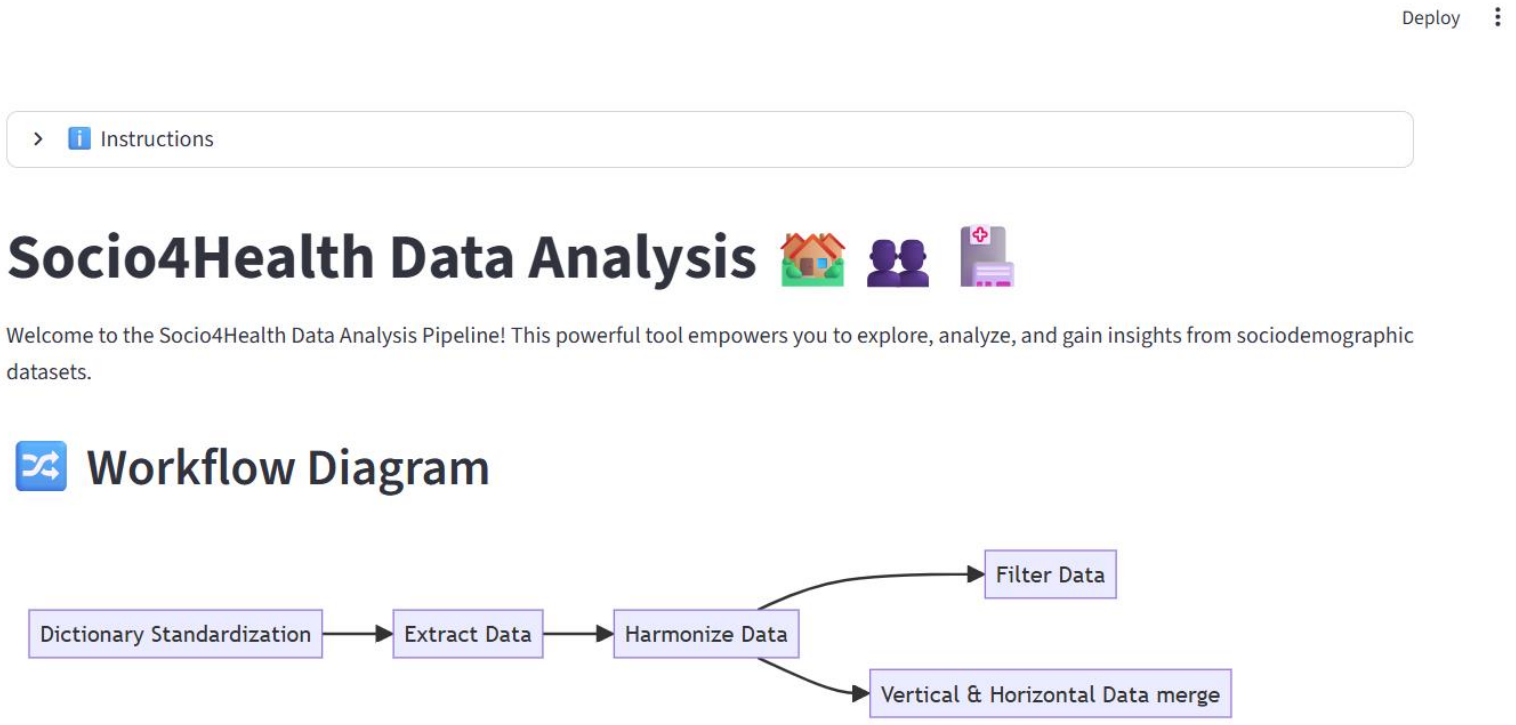
```
pip install -r requirements.txt
```

4. Run the app:

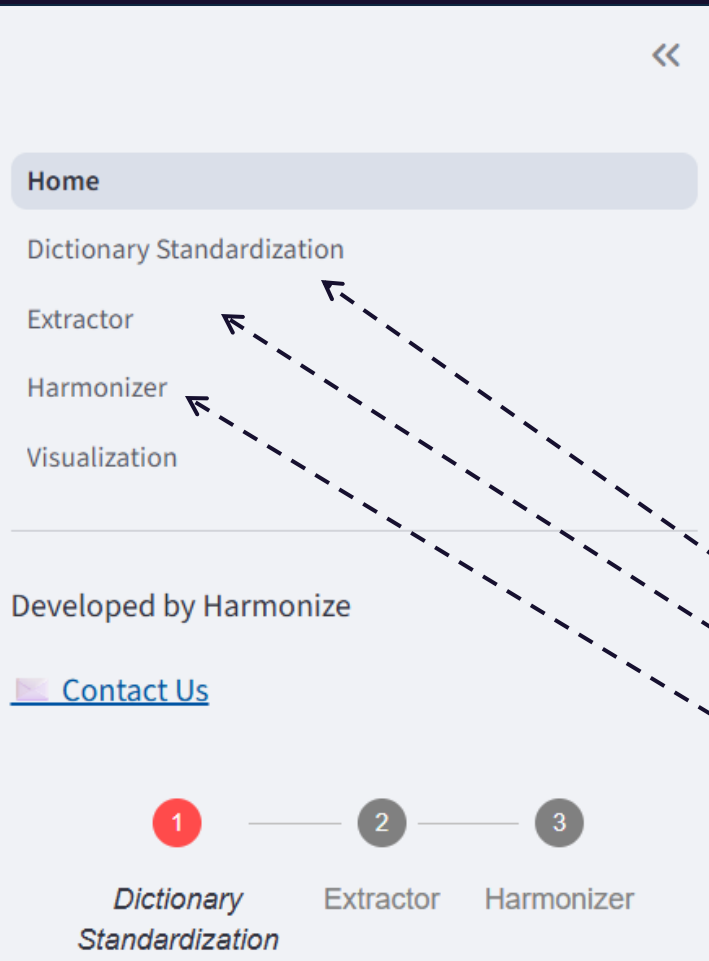
```
streamlit run Home.py --server.maxUploadSize=500
```

How to use socio4health GUI





socio4health



We design a friendly and easy-to-use graphic user interface (GUI) to those less familiarized to coding

Here you can use all functionalities provided by the **socio4health** package

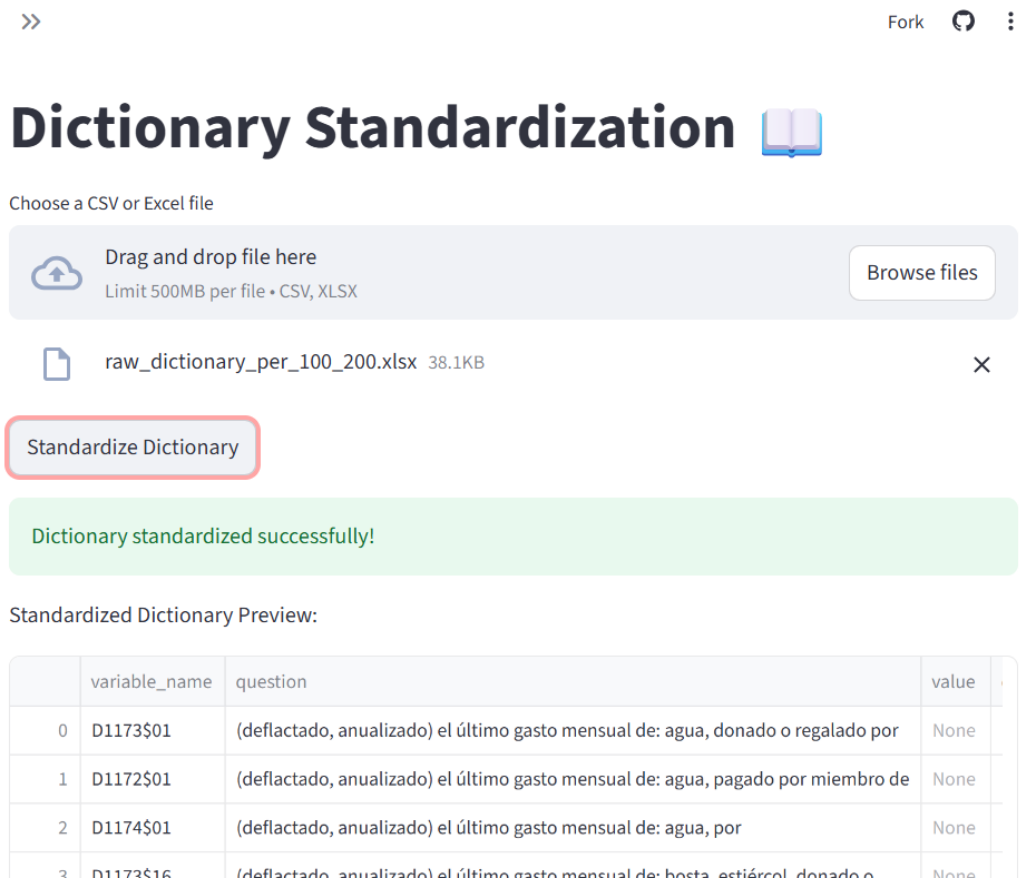
1. Dictionary Standardization
2. Extractor
3. Harmonizer

1. Dictionary Standardization

Follow the instructions on the page to upload your previously created raw dictionary

You can indicate whether this is a FWF file

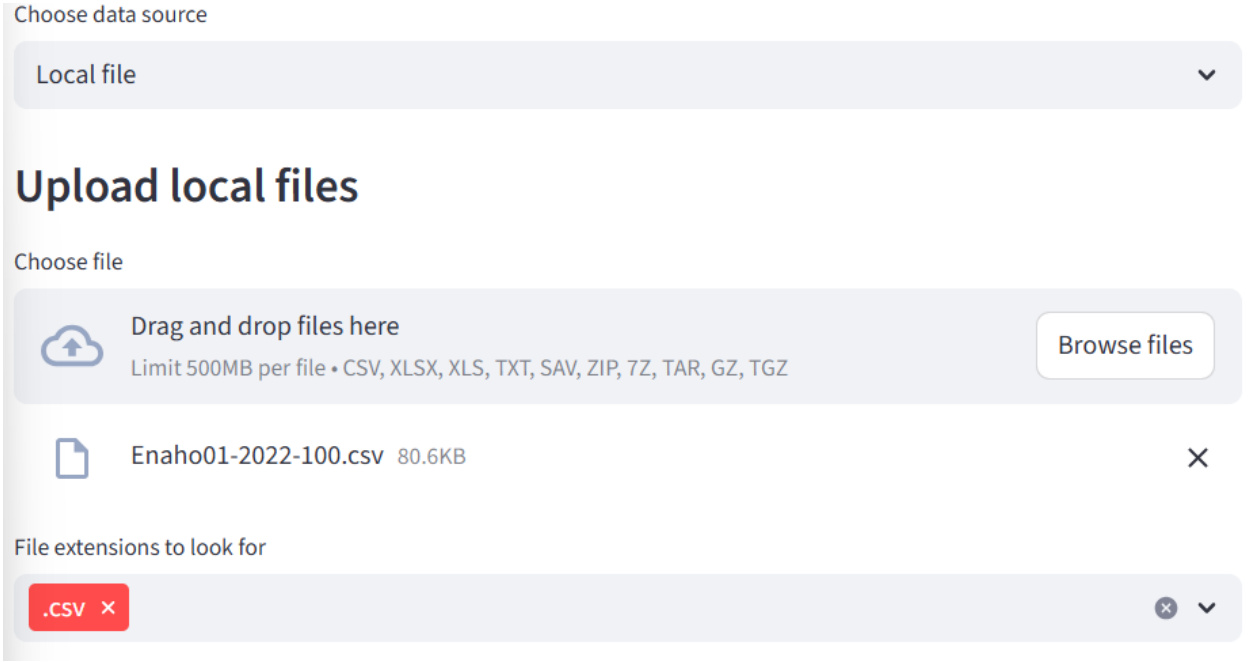
This will load the standardized dictionary directly for the next steps in the **socio4health** data wrangling and analysis pipeline

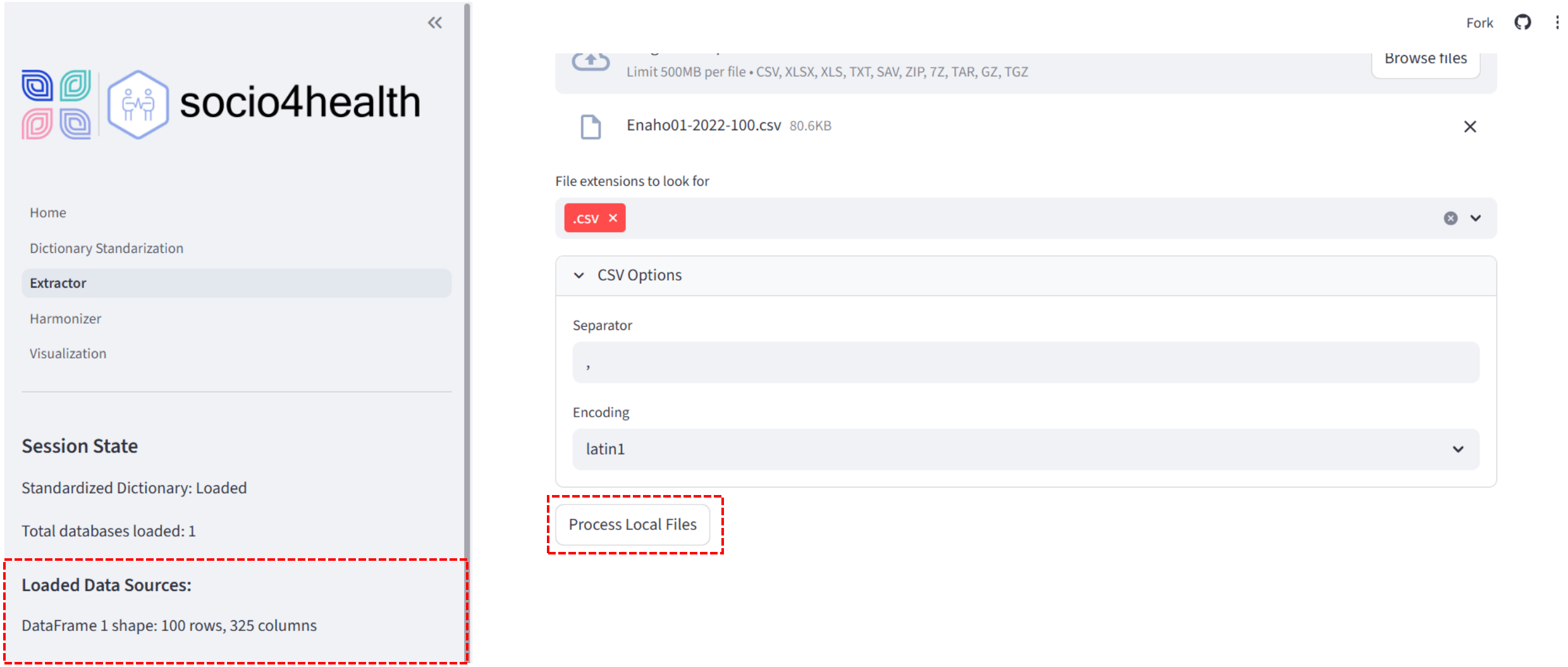


2. Extraction

On the **Extractor** page, click on the dropdown menu and select an Internet URL, a local file or predetermined data

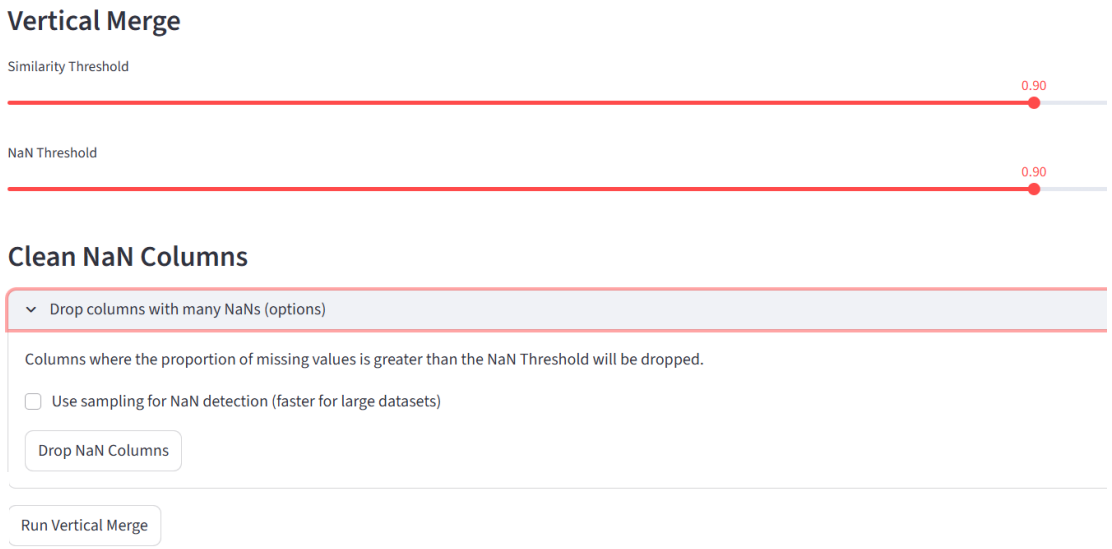
Enter the parameters displayed, if necessary, as well as the CSV options for the extraction

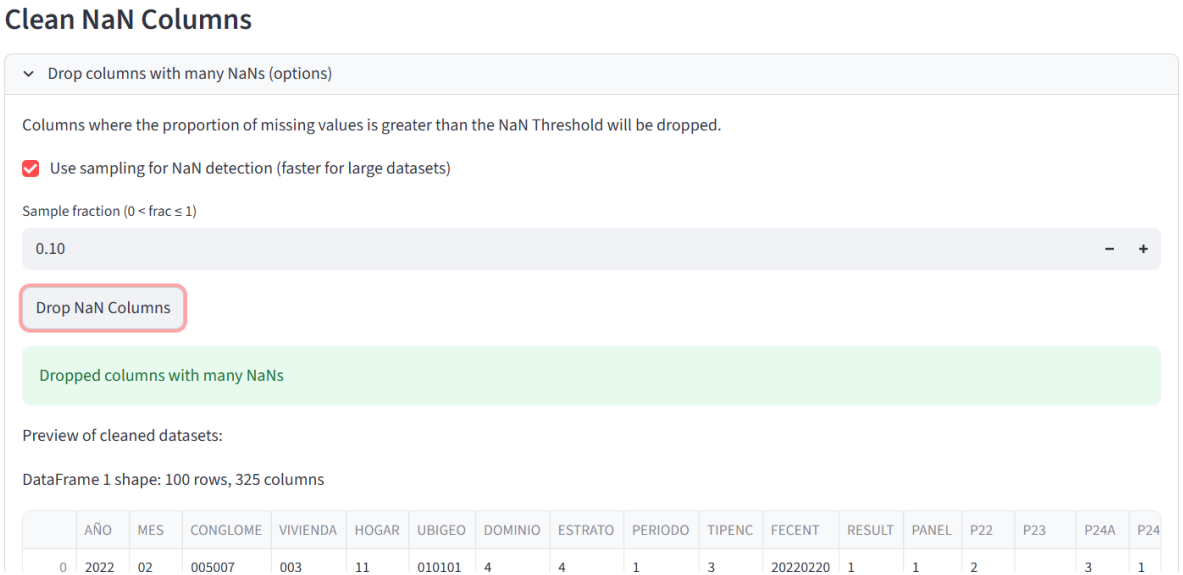



socio4health
Home
Dictionary Standarization
Extractor
Harmonizer
Visualization
Session State
Standardized Dictionary: Loaded
Total databases loaded: 1
Loaded Data Sources:
DataFrame 1 shape: 100 rows, 325 columns
Fork
Limit 500MB per file • CSV, XLSX, XLS, TXT, SAV, ZIP, 7Z, TAR, GZ, TGZ
Browse files
Enaho01-2022-100.csv 80.6KB
File extensions to look for
.csv
CSV Options
Separator
,
Encoding
latin1
Process Local Files

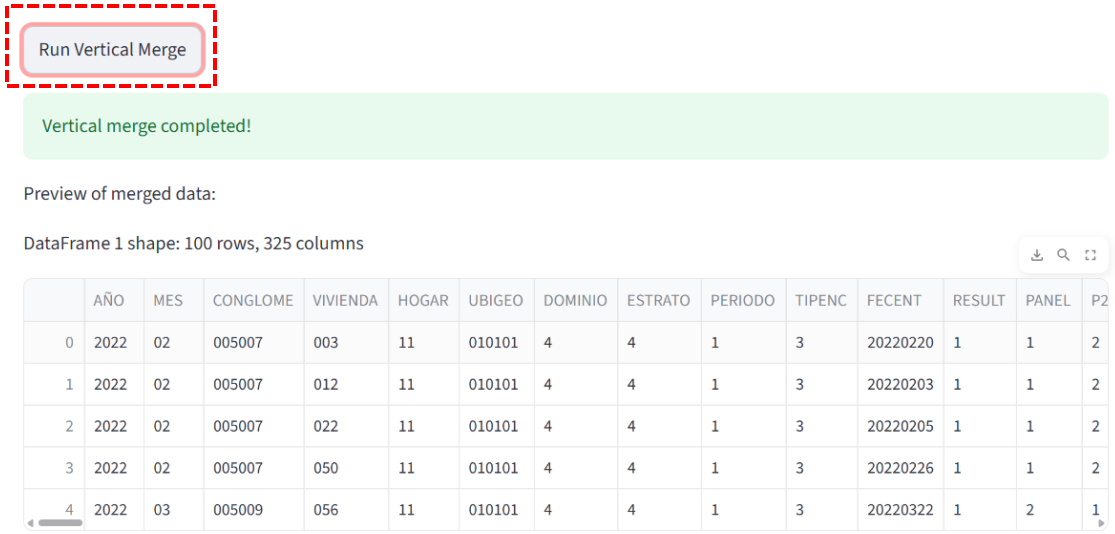
3. Harmonization

On the **Harmoziner** page, drag the slider to adjust the similarity threshold and





Indicate how to handle NaN values from columns by setting the sample fraction and NaN threshold

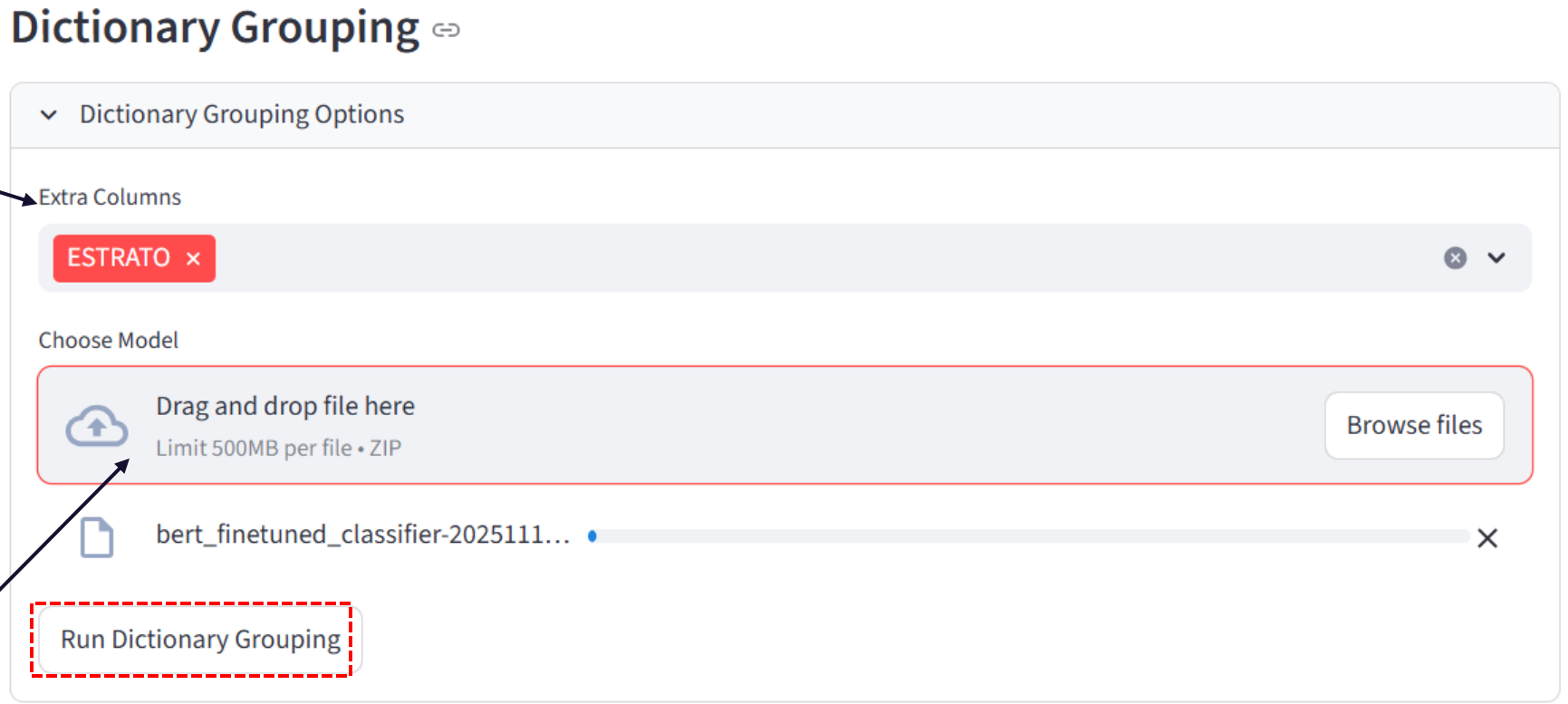


	AÑO	MES	CONGLOME	VIVIENDA	HOGAR	UBIGEO	DOMINIO	ESTRATO	PERIODO	TIPENC	FECENT	RESULT	PANEL	P2
0	2022	02	005007	003	11	010101	4	4	1	3	20220220	1	1	2
1	2022	02	005007	012	11	010101	4	4	1	3	20220203	1	1	2
2	2022	02	005007	022	11	010101	4	4	1	3	20220205	1	1	2
3	2022	02	005007	050	11	010101	4	4	1	3	20220226	1	1	2
4	2022	03	005009	056	11	010101	4	4	1	3	20220322	1	2	1

Click on "**Run Vertical Merge**"

After processing, a preview table of the merged data will be displayed

To categorize the dictionary's rows, select the additional columns (those that differ from the required columns: **question, variable name, value, description,** and **possible answers)**



Drag and drop the file of the pre-trained **BERT model**
Click "**Run Dictionary Grouping**"

Model extracted to bert_model/bert_finetuned_classifier

Model files:

- tokenizer_config.json
- vocab.txt
- config.json
- special_tokens_map.json
- model.safetensors
- tokenizer.json

Using model at: `bert_model/bert_finetuned_classifier`