

OpenAlex Analysis

Generated by Doxygen 1.9.1

1 OpenAlex Analysis	1
1.1 Examples	1
1.1.1 Basic	1
1.1.2 Concepts yearly count	2
1.2 Configure the library	2
1.2.1 Default settings	3
2 Data folder	5
3 Hierarchical Index	7
3.1 Class Hierarchy	7
4 Class Index	9
4.1 Class List	9
5 Class Documentation	11
5.1 openalex_analysis.analysis.entities_analysis.AnalysisConfig Class Reference	11
5.1.1 Detailed Description	12
5.2 openalex_analysis.analysis.entities_analysis.AuthorsAnalysis Class Reference	13
5.3 openalex_analysis.plot.entities_plot.AuthorsPlot Class Reference	14
5.4 openalex_analysis.analysis.entities_analysis.ConceptsAnalysis Class Reference	15
5.5 openalex_analysis.plot.entities_plot.ConceptsPlot Class Reference	16
5.6 openalex_analysis.names.entitie_names.EntitieNames Class Reference	17
5.6.1 Detailed Description	17
5.7 openalex_analysis.analysis.entities_analysis.EntitiesAnalysis Class Reference	17
5.7.1 Detailed Description	19
5.7.2 Constructor & Destructor Documentation	19
5.7.2.1 __init__()	19
5.7.3 Member Function Documentation	20
5.7.3.1 add_average_combined_concept_score_to_multi_concept_entitie_df()	20
5.7.3.2 auto_remove_databases_saved()	20
5.7.3.3 create_multi_concept_filters_entities_dataframe()	20
5.7.3.4 get_api_query()	21
5.7.3.5 get_count_entities_matched()	21
5.7.3.6 get_database_file_name()	21
5.7.3.7 get_df_filtered_entities_selection_threshold()	22
5.7.3.8 get_entitie_string_name()	22
5.7.3.9 get_entitie_type_from_id()	22
5.7.3.10 get_name_of_entitie()	23
5.7.3.11 get_number_of_entities_selected()	23
5.8 openalex_analysis.plot.entities_plot.EntitiesPlot Class Reference	24
5.8.1 Detailed Description	25
5.8.2 Member Function Documentation	25
5.8.2.1 get_figure_entities_of_a_concept_color_country()	25

5.8.2.2 <code>get_figure_entities_selection_threshold()</code>	25
5.8.2.3 <code>get_figure_time_series_element_used_by_entities()</code>	26
5.9 <code>openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis</code> Class Reference	27
5.9.1 Detailed Description	28
5.9.2 Member Function Documentation	28
5.9.2.1 <code>filter_and_format_entitie_data_from_api_response()</code>	28
5.9.2.2 <code>get_sum_concept_scores()</code>	28
5.10 <code>openalex_analysis.plot.entities_plot.InstitutionsPlot</code> Class Reference	29
5.10.1 Detailed Description	30
5.10.2 Member Function Documentation	30
5.10.2.1 <code>get_figure_institutions_multi_concepts_filtered()</code>	30
5.10.2.2 <code>getCustomData()</code>	30
5.10.2.3 <code>getHoverTemplate()</code>	31
5.11 <code>openalex_analysis.analysis.entities_analysis.PublishersAnalysis</code> Class Reference	32
5.12 <code>openalex_analysis.plot.entities_plot.PublishersPlot</code> Class Reference	33
5.13 <code>openalex_analysis.analysis.entities_analysis.SourcesAnalysis</code> Class Reference	34
5.14 <code>openalex_analysis.plot.entities_plot.SourcesPlot</code> Class Reference	35
5.15 <code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> Class Reference	36
5.15.1 Detailed Description	37
5.15.2 Member Function Documentation	38
5.15.2.1 <code>add_statistics_to_element_count_array()</code>	38
5.15.2.2 <code>create_element_used_count_array()</code>	38
5.15.2.3 <code>filter_and_format_entitie_data_from_api_response()</code>	39
5.15.2.4 <code>get_country_code()</code>	39
5.15.2.5 <code>get_institution_name()</code>	39
5.15.2.6 <code>get_works_concepts_count()</code>	40
5.15.2.7 <code>get_works_references_count()</code>	40
5.15.2.8 <code>sort_count_array()</code>	40
5.16 <code>openalex_analysis.plot.entities_plot.WorksPlot</code> Class Reference	41
5.16.1 Detailed Description	41
5.16.2 Member Function Documentation	41
5.16.2.1 <code>get_figure_nb_time_referenced()</code>	41
5.16.2.2 <code>getCustomData()</code>	42
5.16.2.3 <code>getHoverTemplate()</code>	42

Chapter 1

OpenAlex Analysis

Scientific literature analysis using the OpenAlex API.

This repo provides classes and methods to extract statistics, plots and graphs, as well as examples in Jupyter Notebooks.

Install with:

```
pip install openalex-analysis
```

A web app based on the library is available [here](#).

Documentation : <https://romain894.github.io/openalex-analysis>

[OpenAlex documentation](#) [Explore OpenAlex in a GUI](#)

Licence: GPL V3

1.1 Examples

More examples can be found in the notebooks [Works_examples.ipynb](#) and [Concepts_Works_analysis.ipynb](#)

1.1.1 Basic

In the example, we create a dataset with the works about sustainability.

This dataset can be used as it, it is stored in a parquet file (more optimized than CSV) on the computer and can be simply imported as a dataframe with Pandas.

After getting this dataset, we continue by extracting the most cited articles by the dataset. For that, we extract all the references of the articles present in the dataset and rank these references.

```
from openalex_analysis.plot import WorksPlot
concept_sustainability_id = 'C66204764'
# get the works about sustainability
wplt = WorksPlot(concept_sustainability_id)
print("\nFirst entities in the dataset:")
print(wplt.entities_df[['id', 'title']].head(3))
# compute the most cited works by the dataset previously downloaded
wplt.create_element_used_count_array('reference')
print("\nMost cited work within the dataset:")
print(wplt.element_count_df.head(3))
```

```

Loading dataframe of works of the concept C66204764
Loading the list of entities from a parquet file...
First entities in the dataset:
      id                                     title
0  https://openalex.org/W2101946146  Asset Stock Accumulation and Sustainability of...
1  https://openalex.org/W1999167944  Planetary boundaries: Guiding human developmen...
2  https://openalex.org/W2122266551  Agricultural sustainability and intensive prod...
Getting name of C66204764 from the OpenAlex API (cache disabled)...
Creating the works references count of works C66204764...
Most cited work within the dataset:
      C66204764 Sustainability
element
https://openalex.org/W2026816730      262
https://openalex.org/W2096885696      249
https://openalex.org/W2103847341      203

```

1.1.2 Concepts yearly count

In this example, we will create two datasets: one with the articles about sustainability of the SRC (Stockholm Resilience Centre) and one with the articles about sustainability of the UTT (University of Technology of Troyes).

We will then plot the yearly usage of the concept sustainability by these institutions (in this case it's equal to the number of articles in the dataset, as the dataset contains only the articles about sustainability).

We could also plot the yearly usage of other concepts or of the references by changing the parameters of the functions `create_element_used_count_array()` and `get_figure_time_series_element_used_by_entities()`.

```

from openalex_analysis.plot import InstitutionsPlot, WorksPlot
concept_sustainability_id = 'C66204764'
# create the filter for the API to get only the articles about sustainability
sustainability_concept_filter = {"concepts": {"id": concept_sustainability_id}}
# set the years we want to count
count_years = list(range(2004, 2024))
institution_ids_list = ["I138595864", "I140494188"]
institution_names_list = ["Stockholm Resilience Centre", "University of Technology of Troyes"]
# create a list of dictionaries with each dictionary containing the ID, name and filter for each institution
entities_ref_to_count = [None] * len(institution_ids_list)
for i in range(len(institution_ids_list)):
    entities_ref_to_count[i] = {'entity_from_id': institution_ids_list[i],
                              'extra_filters': sustainability_concept_filter,
                              'entity_name': institution_names_list[i]}

wplt = WorksPlot()
wplt.create_element_used_count_array('concept', entities_ref_to_count, count_years = count_years)
wplt.add_statistics_to_element_count_array(sort_by = 'sum_all_entities', min_concept_level = 2)
wplt.get_figure_time_series_element_used_by_entities().write_image("Plot_yearly_usage_sustainability_SRC_UTT.svg",
    width=1200)
wplt.get_figure_time_series_element_used_by_entities()

```

1.2 Configure the library

By default, the library will run out of the box. Nevertheless, some optional configurations can be done to improve the performance and to fit best the use case.

Setting up the email address allows you to use the polite pool from OpenAlex which is faster than the default one.

```

from openalex_analysis.plot import config, InstitutionsPlot
config.email = "email@example.com"
InstitutionsPlot()

```

The notebook [Setup_example.ipynb](#) contains more setup examples.

1.2.1 Default settings

```
config.email = None
config.api_key = None
config.openalex_url = "https://api.openalex.org"
config.allow_automatic_download = True
config.disable_tqdm_loading_bar = False
config.n_max_entities = 10000
config.project_datas_folder_path = "data"
config.parquet_compression = "brotli"
config.max_storage_percent = 95
config.redis_enabled = False
# Uncomment the following lines if you want to use Redis cache
# config.redis_client = StrictRedis(host=os.environ.get('DOCKER_REDIS_URL', "localhost"),
#                                   decode_responses=True,
#                                   port=6379,
#                                   db=2,)
# config.redis_cache = RedisCache(redis_client=config_redis_client)
# Don't forget to add the following two lines with all the imports
# from redis import StrictRedis
# from redis_cache import RedisCache
```

- `email` The email address is need to access the polite pool from OpenAlex which is faster than the default one.
- `api_key` Optional, if you have one from OpenAlex
- `openalex_url` OpenAlex URL
- `allow_automatic_download` Allow the library to download dataset from OpenAlex if not already present on the disk
- `disable_tqdm_loading_bar` If set to True, it will disable the loading bar in the terminal output when downloading data from the OpenAlex API.
- `n_max_entities` When downloading a list of entities from the API (eg a list of works), the maximum number of entities to download. Set to None to have no limitation. This number must be a multiple of 200 (the is the number of element per page used by the library)
- `project_datas_folder_path` Path to store the data downloaded from the API. The data will be stored as parquet files, with each file corresponding to one request.
- `parquet_compression` By default, the parquet files are compressed. The compression can be disabled by setting with `parquet_compression = None`. For other parquet compression algorithms, see the pandas documentation. Compressing reduces by 2 to 10 the file size while needing a negligeable time to compress or decompress. Disabling the compression is usefull if you want to read the parquet files with an external software.
- `max_storage_percent` Maximum storage usage percentage on the disk before starting to delete data stored in `project_datas_folder_path`. The parquet file with the oldest last read data will be deleted first.
- `redis_enabled` Whenever Redis cache is enabled or not
- `redis_client` The Redis client configuration. Don't forget to add `from redis import StrictRedis` where the configuration is defined.
- `redis_cache` The Redis cache configuration. Don't forget to add `from redis_cache import RedisCache` where the configuration is defined.

Romain Thomas 2023

Chapter 2

Data folder

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

dict	
openalex_analysis.analysis.entities_analysis.AnalysisConfig	11
openalex_analysis.names.entitie_names.EntityNames	17
openalex_analysis.plot.entities_plot.EntitiesPlot	24
openalex_analysis.plot.entities_plot.AuthorsPlot	14
openalex_analysis.plot.entities_plot.ConceptsPlot	16
openalex_analysis.plot.entities_plot.InstitutionsPlot	29
openalex_analysis.plot.entities_plot.PublishersPlot	33
openalex_analysis.plot.entities_plot.SourcesPlot	35
openalex_analysis.plot.entities_plot.WorksPlot	41
Authors	
openalex_analysis.analysis.entities_analysis.AuthorsAnalysis	13
openalex_analysis.plot.entities_plot.AuthorsPlot	14
Concepts	
openalex_analysis.analysis.entities_analysis.ConceptsAnalysis	15
openalex_analysis.plot.entities_plot.ConceptsPlot	16
EntityNames	
openalex_analysis.analysis.entities_analysis.EntitiesAnalysis	17
openalex_analysis.analysis.entities_analysis.AuthorsAnalysis	13
openalex_analysis.analysis.entities_analysis.ConceptsAnalysis	15
openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis	27
openalex_analysis.plot.entities_plot.InstitutionsPlot	29
openalex_analysis.analysis.entities_analysis.PublishersAnalysis	32
openalex_analysis.plot.entities_plot.PublishersPlot	33
openalex_analysis.analysis.entities_analysis.SourcesAnalysis	34
openalex_analysis.plot.entities_plot.SourcesPlot	35
openalex_analysis.analysis.entities_analysis.WorksAnalysis	36
openalex_analysis.plot.entities_plot.WorksPlot	41
Institutions	
openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis	27
Publishers	
openalex_analysis.analysis.entities_analysis.PublishersAnalysis	32
Sources	
openalex_analysis.analysis.entities_analysis.SourcesAnalysis	34
Works	
openalex_analysis.analysis.entities_analysis.WorksAnalysis	36

Chapter 4

Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

openalex_analysis.analysis.entities_analysis.AnalysisConfig	
TODO	11
openalex_analysis.analysis.entities_analysis.AuthorsAnalysis	13
openalex_analysis.plot.entities_plot.AuthorsPlot	14
openalex_analysis.analysis.entities_analysis.ConceptsAnalysis	15
openalex_analysis.plot.entities_plot.ConceptsPlot	16
openalex_analysis.names.entitie_names.EntitieNames	17
openalex_analysis.analysis.entities_analysis.EntitiesAnalysis	
OpenAlexAnalysis class which contains generic methods to do analysis over OpenAlex entities	17
openalex_analysis.plot.entities_plot.EntitiesPlot	
EntitiesPlot class which contains generic methods to do plots of OpenAlex entities	24
openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis	
This class contains specific methods for Institutions concepts analysis	27
openalex_analysis.plot.entities_plot.InstitutionsPlot	
This class contains specific methods for Institutions concepts plot	29
openalex_analysis.analysis.entities_analysis.PublishersAnalysis	32
openalex_analysis.plot.entities_plot.PublishersPlot	33
openalex_analysis.analysis.entities_analysis.SourcesAnalysis	34
openalex_analysis.plot.entities_plot.SourcesPlot	35
openalex_analysis.analysis.entities_analysis.WorksAnalysis	
This class contains specific methods for Works concepts analysis	36
openalex_analysis.plot.entities_plot.WorksPlot	
This class contains specific methods for Works concepts plot	41

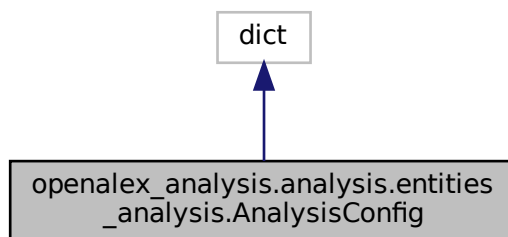
Chapter 5

Class Documentation

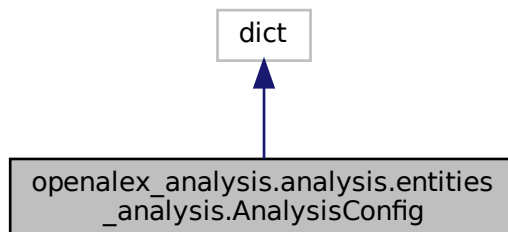
5.1 openalex_analysis.analysis.entities_analysis.AnalysisConfig Class Reference

TODO.

Inheritance diagram for openalex_analysis.analysis.entities_analysis.AnalysisConfig:



Collaboration diagram for openalex_analysis.analysis.entities_analysis.AnalysisConfig:



Public Member Functions

- def `__getattr__` (self, key)
- def `__setattr__` (self, key, value)

5.1.1 Detailed Description

TODO.

Parameters

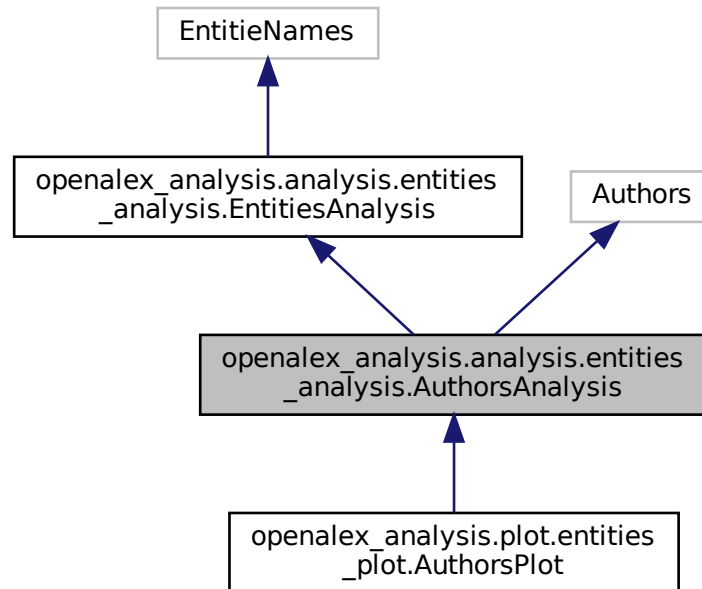
<i>email</i>	Email
<i>api_key</i>	API key
<i>openalex_url</i>	OpenAlex URL
<i>http_retry_times</i>	HTTP retry times
<i>allow_automatic_download</i>	The allow automatic download (True/False)
<i>disable_tqdm_loading_bar</i>	The disable tqdm loading bar (True/False)
<i>n_max_entities</i>	TODO
<i>project_datas_folder_path</i>	TODO
<i>parquet_compression</i>	TODO
<i>max_storage_percent</i>	TODO
<i>redis_parameters</i>	TODO TO UPDATE DOC

The documentation for this class was generated from the following file:

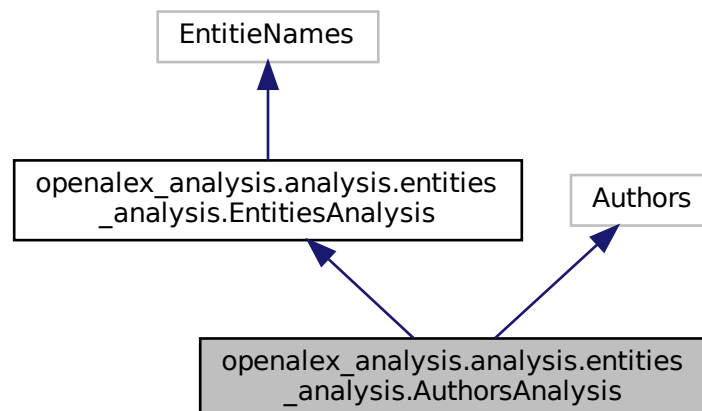
- `openalex_analysis/analysis/entities_analysis.py`

5.2 openalex_analysis.analysis.entities_analysis.AuthorsAnalysis Class Reference

Inheritance diagram for openalex_analysis.analysis.entities_analysis.AuthorsAnalysis:



Collaboration diagram for openalex_analysis.analysis.entities_analysis.AuthorsAnalysis:



Static Public Attributes

- **EntitieOpenAlex** = Authors

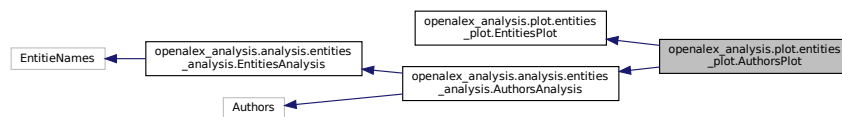
Additional Inherited Members

The documentation for this class was generated from the following file:

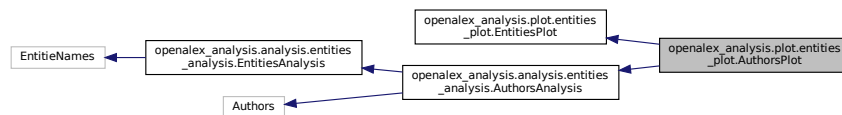
- `openalex_analysis/analysis/entities_analysis.py`

5.3 `openalex_analysis.plot.entities_plot.AuthorsPlot` Class Reference

Inheritance diagram for `openalex_analysis.plot.entities_plot.AuthorsPlot`:



Collaboration diagram for `openalex_analysis.plot.entities_plot.AuthorsPlot`:



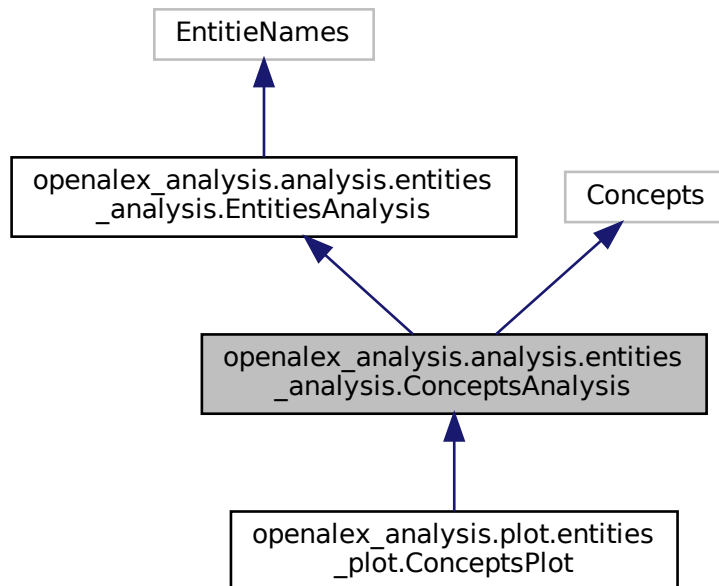
Additional Inherited Members

The documentation for this class was generated from the following file:

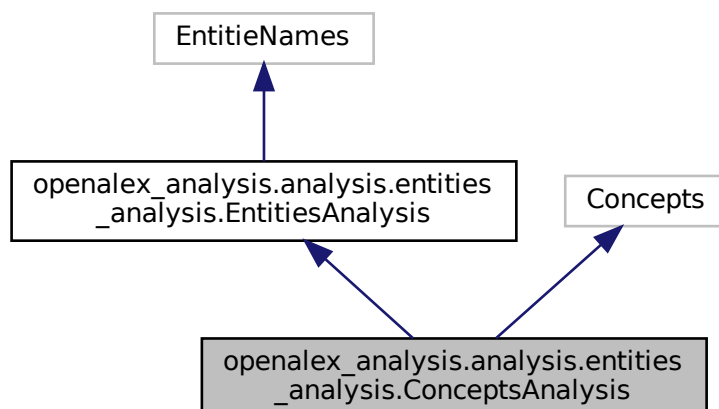
- `openalex_analysis/plot/entities_plot.py`

5.4 openalex_analysis.analysis.entities_analysis.ConceptsAnalysis Class Reference

Inheritance diagram for openalex_analysis.analysis.entities_analysis.ConceptsAnalysis:



Collaboration diagram for openalex_analysis.analysis.entities_analysis.ConceptsAnalysis:



Static Public Attributes

- **EntitieOpenAlex** = Concepts

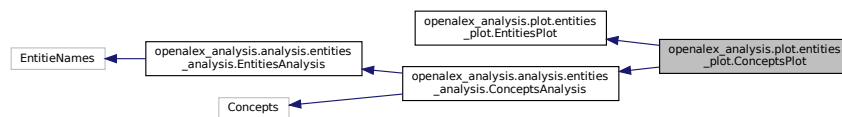
Additional Inherited Members

The documentation for this class was generated from the following file:

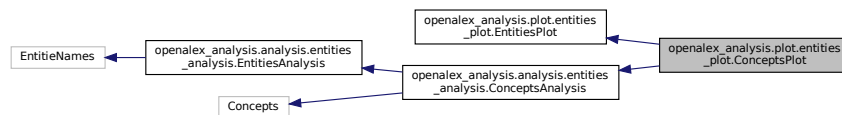
- `openalex_analysis/analysis/entities_analysis.py`

5.5 `openalex_analysis.plot.entities_plot.ConceptsPlot` Class Reference

Inheritance diagram for `openalex_analysis.plot.entities_plot.ConceptsPlot`:



Collaboration diagram for `openalex_analysis.plot.entities_plot.ConceptsPlot`:



Additional Inherited Members

The documentation for this class was generated from the following file:

- `openalex_analysis/plot/entities_plot.py`

5.6 openalex_analysis.names.entity_names.EntityNames Class Reference

Static Public Attributes

- string `concepts_parquet_file_path` = "list_all_concepts.parquet"
`CONCEPTS #####.`
- `concepts_df` = `pd.read_parquet(BytesIO(pkgutil.get_data(__name__, concepts_parquet_file_path)))`
- `concepts_names` = `concepts_df[['openalex_id', 'display_name']].set_index('openalex_id')['display_name'].to_dict()`
- `concepts_levels` = `concepts_df[['openalex_id', 'level']].set_index('openalex_id')['level'].to_dict()`
- `concepts_normalized_names` = `concepts_df[['openalex_id', 'normalized_name']].set_index('openalex_id')['normalized_name'].to_dict()`
- string `institutions_parquet_file_path` = "list_all_institutions.parquet"
`INSTITUTIONS #####.`
- `institutions_df` = `pd.read_parquet(BytesIO(pkgutil.get_data(__name__, institutions_parquet_file_path)))`
- `institutions_names` = `institutions_df[['id', 'display_name']].set_index('id')['display_name'].to_dict()`

5.6.1 Detailed Description

Class to manage the concept and institution names

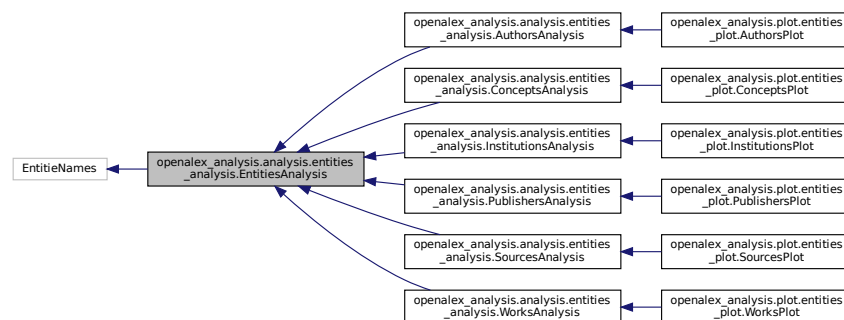
The documentation for this class was generated from the following file:

- `openalex_analysis/names/entity_names.py`

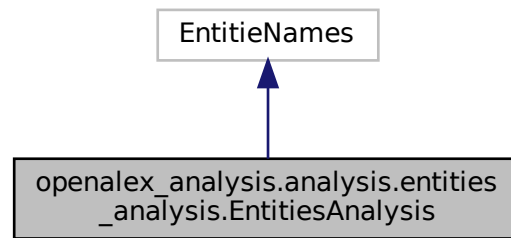
5.7 openalex_analysis.analysis.entities_analysis.EntitiesAnalysis Class Reference

OpenAlexAnalysis class which contains generic methods to do analysis over OpenAlex entities.

Inheritance diagram for `openalex_analysis.analysis.entities_analysis.EntitiesAnalysis`:



Collaboration diagram for `openalex_analysis.analysis.entities_analysis.EntitiesAnalysis`:



Public Member Functions

- `def __init__ (self, entitie_from_id=None, extra_filters=None, database_file_path=None, create_dataframe=True, entitie_name=None, load_only_columns=None)`
- `def get_count_entities_matched (self, query_filters)`
Gets and return the number of entities which match the query fitlers.
- `def get_api_query (self)`
Gets the api query from the parameters of the instance.
- `def download_list_entities (self)`
Downloads the entities which match the parameters of the instance, and store the dataset as a parquet file.
- `def load_entities_dataframe (self)`
Loads an entities dataset from file (or download it if needed and allowed by the instance) to the dataframe of the instance.
- `def auto_remove_databases_saved (self)`
Remove databases files (the data downloaded from OpenAlex) if the storage is full.
- `def get_df_filtered_entities_selection_threshold (self, df_filters)`
Gets df_filtered which contains the entities of self.entities_df fitting the filters in df_filters.
- `def get_number_of_entities_selected (self, x_threshold, y_threshold, cited_by_threshold, x_datas, y_datas)`
Gets the number of entities selected on the plot.
- `def create_multi_concept_filters_entities_dataframe (self, concepts_from, concepts_filters, thresholds, x_datas, x_threshold, cited_by_threshold)`
Creates the multi concept filters entities dataframe.
- `def add_average_combined_concept_score_to_multi_concept_entitie_df (self, concepts_from)`
Adds a column with the average combined concept score to the multi concept entities dataframe.
- `def get_database_file_name (self, entitie_from_id=None, entities_type=None, db_format="parquet", extra_text=None)`
Gets the database file name according to the parameters of the object or the arguments given.
- `def get_entitie_string_name (self, entitie=None)`
Gets the entitie type string name.
- `def get_entitie_type_from_id (self, entitie=None)`
Gets the entitie type from the entitie id.
- `def get_name_of_entitie (self, entitie=None, allow_download_from_API=True)`
Gets the name of entitie.
- `def get_info_about_entitie (self, entitie, infos=["display_name"], return_as_pd_serie=True, allow_download_from_API=True)`

Public Attributes

- `per_page`
- `entitie_from_id`
- `entitie_from_type`
- `extra_filters`
- `database_file_path`
- `entitie_name`
- `load_only_columns`
- `entities_df`
- `entities_multi_filtered_df`
- `element_count_df`
- `entitie_downloading_progress_percentage`
- `create_element_count_array_progress_percentage`
- `create_element_count_array_progress_text`
- `count_element_type`
- `count_element_years`
- `count_entities_cols`

Static Public Attributes

- `cc = coco.CountryConverter()`

5.7.1 Detailed Description

OpenAlexAnalysis class which contains generic methods to do analysis over OpenAlex entities.

5.7.2 Constructor & Destructor Documentation

5.7.2.1 `__init__()`

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.__init__ (
    self,
    entitie_from_id = None,
    extra_filters = None,
    database_file_path = None,
    create_dataframe = True,
    entitie_name = None,
    load_only_columns = None )
```

Parameters

<i>entitie_from_id</i>	The entitie identifier (eg an institution id) from which to take the entities (eg the works) to analyse (str)
<i>filters</i>	Optional additionnal filters, refer to the documentation of openalex and pyalex for the format (dict)
<i>database_file_path</i>	The database file path to force the analyse over datas in a specific file (str)
<i>create_dataframe</i>	Create the dataframe at the initialisation (and download the data if needed and allowed)
<i>entitie_name</i>	To specify the name of the entitie to avoid downloading it via the API if needed

5.7.3 Member Function Documentation

5.7.3.1 add_average_combined_concept_score_to_multi_concept_entitie_df()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.add_average_combined_↵
concept_score_to_multi_concept_entitie_df (
    self,
    concepts_from )
```

Adds a column with the average combined concept score to the multi concept entities dataframe.

Parameters

<i>concepts_from</i>	The concepts to use to calculate the combined concept score (list of str)
----------------------	---

5.7.3.2 auto_remove_databases_saved()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.auto_remove_databases_saved
(
    self )
```

Remove databases files (the data downloaded from OpenAlex) if the storage is full.

It keeps the last accessed files

5.7.3.3 create_multi_concept_filters_entities_dataframe()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.create_multi_concept_↵
filters_entities_dataframe (
    self,
    concepts_from,
    concepts_filters,
    thresholds,
    x_datas,
    x_threshold,
    cited_by_threshold )
```

Creates the multi concept filters entities dataframe.

Combines different datasets and filters them.

Parameters

<i>concepts_from</i>	The concept datasets to import and on which the filters will be applied (list of str)
<i>concepts_filters</i>	The concepts which will be used to filter (list of str)
<i>thresholds</i>	The thresholds attached to each concepts to filter (list of float or int)
<i>x_datas</i>	The dataframe key of the global filter (eg the number of works) (str)
<i>x_threshold</i>	The threshold for the global filter (float or int)
<i>cited_by_threshold</i>	The cited by threshold (another global filter) (float or int)

5.7.3.4 get_api_query()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_api_query (
    self )
```

Gets the api query from the parameters of the instance.

Returns

The api query (dict)

5.7.3.5 get_count_entities_matched()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_count_entities_matched (
    self,
    query_filters )
```

Gets and return the number of entities which match the query filters.

Parameters

<i>query_filters</i>	The query filters (dict)
----------------------	--------------------------

Returns

The count entities matched (int)

5.7.3.6 get_database_file_name()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_database_file_name (
    self,
    entitie_from_id = None,
    entities_type = None,
    db_format = "parquet",
    extra_text = None )
```

Gets the database file name according to the parameters of the object or the arguments given.

Parameters

<i>entitie_from_id</i>	The identifier of the entitie (eg a concept id) which was used to filter the entities (eg works) in the database (str)
<i>entities_type</i>	The entities type in the database (eg works) (EntitieOpenAlex)
<i>db_format</i>	The database file format (str)
<i>extra_text</i>	Extra text to add to the file name (str)

Returns

The database file name (str)

5.7.3.7 get_df_filtered_entities_selection_threshold()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_df_filtered_entities_↵
selection_threshold (
    self,
    df_filters )
```

Gets df_filtered which contains the entities of self.entities_df fitting the filters in df_filters.

Parameters

<i>df_filters</i>	The filters in a dictionary with for the key for the data to filter and for the value the minimum threshold (dict)
-------------------	--

Returns

df_filtered, corresponding the the entities fitting the thresholds (pandas DataFrame)

5.7.3.8 get_entitie_string_name()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_entitie_string_name (
    self,
    entitie = None )
```

Gets the entitie type string name.

Parameters

<i>entitie</i>	The entitie, if not provided, the instance entitie id will be used (BaseOpenAlex)
----------------	---

Returns

The entitie type name (str)

5.7.3.9 get_entitie_type_from_id()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_entitie_type_from_id (
    self,
    entitie = None )
```

Gets the entitie type from the entitie id.

Parameters

<i>entitie</i>	The entitie id (str)
----------------	----------------------

Returns

The entitie type (BaseOpenAlex)

5.7.3.10 get_name_of_entitie()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_name_of_entitie (
    self,
    entitie = None,
    allow_download_from_API = True )
```

Gets the name of entitie.

Parameters

<i>entitie</i>	The entitie id, if not provided, use the one from the instance (str)
<i>allow_download_from_API</i>	Allow to download the entitie name from the OpenAlex API (bool)

Returns

The name of entitie (str)

5.7.3.11 get_number_of_entities_selected()

```
def openalex_analysis.analysis.entities_analysis.EntitiesAnalysis.get_number_of_entities_↵
selected (
    self,
    x_threshold,
    y_threshold,
    cited_by_threshold,
    x_datas,
    y_datas )
```

Gets the number of entities selected on the plot.

Parameters

<i>x_threshold</i>	The x threshold (float or int)
<i>y_threshold</i>	The y threshold (float or int)
<i>cited_by_threshold</i>	The cited by threshold (float or int)
<i>x_datas</i>	The x datas key on the dataframe (str)
<i>y_datas</i>	The y datas key on the dataframe (str)

Returns

The number of entities selected (int)

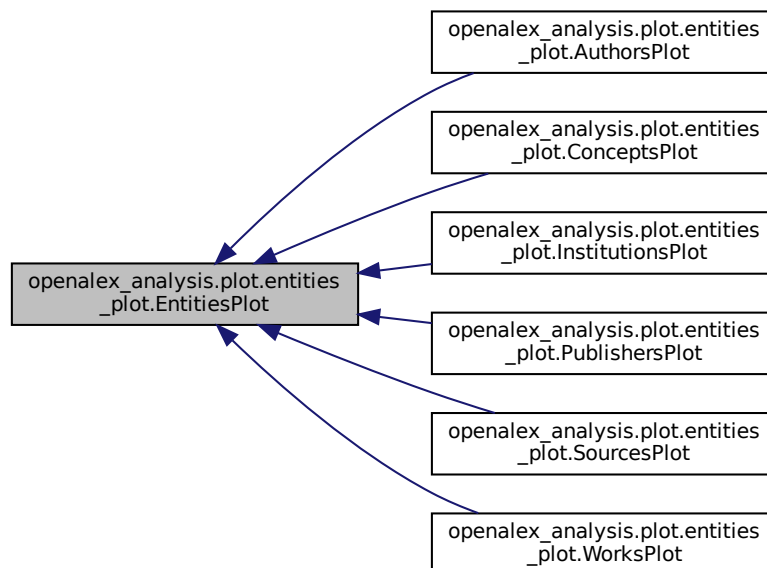
The documentation for this class was generated from the following file:

- `openalex_analysis/analysis/entities_analysis.py`

5.8 `openalex_analysis.plot.entities_plot.EntitiesPlot` Class Reference

`EntitiesPlot` class which contains generic methods to do plots of OpenAlex entities.

Inheritance diagram for `openalex_analysis.plot.entities_plot.EntitiesPlot`:



Public Member Functions

- def `get_figure_entities_of_a_concept_color_country` (self, concept, plot_parameters=None)
Gets the figure with the entities of a concept, and with the country as color.
- def `get_figure_entities_selection_threshold` (self, concept, plot_parameters, x_threshold=0, y_threshold=0, cited_by_threshold=0, display_only_selected_entities=None, display_threshold_lines=None, entity_to_highlight=None)
Gets the figure with the entities of a concept and the selection threshold lines (optional)
- def `get_figure_time_series_element_used_by_entities` (self, element=None, plot_title=None, x_datas='year', x_legend="Year", y_datas=None, color_legend="Entities")
Gets the figure with the time series usage of a element (eg reference, concept) by entities.

5.8.1 Detailed Description

[EntitiesPlot](#) class which contains generic methods to do plots of OpenAlex entities.

5.8.2 Member Function Documentation

5.8.2.1 `get_figure_entities_of_a_concept_color_country()`

```
def openalex_analysis.plot.entities_plot.EntitiesPlot.get_figure_entities_of_a_concept_color←
_country (
    self,
    concept,
    plot_parameters = None )
```

Gets the figure with the entities of a concept, and with the country as color.

Parameters

<i>concept</i>	The concept (str)
<i>plot_parameters</i>	The plot parameters (dict)

Returns

The figure (fig)

5.8.2.2 `get_figure_entities_selection_threshold()`

```
def openalex_analysis.plot.entities_plot.EntitiesPlot.get_figure_entities_selection_threshold
(
    self,
    concept,
    plot_parameters,
    x_threshold = 0,
    y_threshold = 0,
    cited_by_threshold = 0,
    display_only_selected_entities = None,
    display_threshold_lines = None,
    entity_to_highlight = None )
```

Gets the figure with the entities of a concept and the selection threshold lines (optional)

Parameters

<i>concept</i>	The concept (str)
<i>plot_parameters</i>	The plot parameters (dict)

Parameters

<i>x_threshold</i>	The x threshold (float or int)
<i>y_threshold</i>	The y threshold (float or int)
<i>cited_by_threshold</i>	The cited by threshold (float or int)
<i>display_only_selected_entities</i>	The display only selected entities (bool)
<i>display_threshold_lines</i>	The display threshold lines (bool)
<i>entity_to_highlight</i>	The entity to highlight on the plot (str)

Returns

The figure (fig)

5.8.2.3 get_figure_time_series_element_used_by_entities()

```
def openalex_analysis.plot.entities_plot.EntitiesPlot.get_figure_time_series_element_used_by←
_entities (
    self,
    element = None,
    plot_title = None,
    x_datas = 'year',
    x_legend = "Year",
    y_datas = None,
    color_legend = "Entities" )
```

Gets the figure with the time series usage of a element (eg reference, concept) by entities.

Parameters

<i>element</i>	The element (default first in the dataframe) (str)
<i>plot_title</i>	The plot title (str)
<i>x_datas</i>	The x datas (default: year) (str)
<i>x_legend</i>	The x legend (str)
<i>y_datas</i>	The y datas (the entities to plot, the default is all entities in the dataframe) (list[str])
<i>color_legend</i>	The color legend (str)

Returns

The figure (fig)

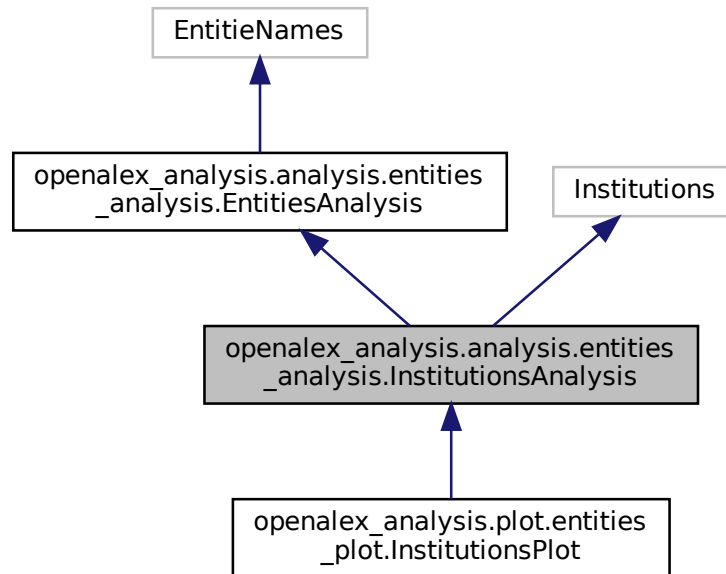
The documentation for this class was generated from the following file:

- openalex_analysis/plot/entities_plot.py

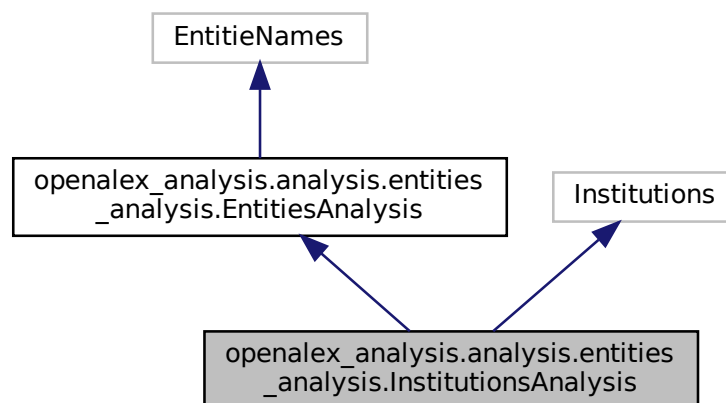
5.9 openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis Class Reference

This class contains specific methods for Institutions concepts analysis.

Inheritance diagram for openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis:



Collaboration diagram for openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis:



Public Member Functions

- def `filter_and_format_entitie_data_from_api_response` (self, entitie)
Filter and format the institutions data downloaded from the API.
- def `get_sum_concept_scores` (self, institutions, concept_links)
Gets the sum of the concept scores of the concepts in the list concepts.

Public Attributes

- `entitie_from_type`

Static Public Attributes

- `EntitieOpenAlex` = Institutions

5.9.1 Detailed Description

This class contains specific methods for Institutions concepts analysis.

5.9.2 Member Function Documentation

5.9.2.1 `filter_and_format_entitie_data_from_api_response()`

```
def openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis.filter_and_format_↵
entitie_data_from_api_response (
    self,
    entitie )
```

Filter and format the institutions data downloaded from the API.

Parameters

<code>entitie</code>	The institutions data from the API (dict)
----------------------	---

Returns

The institutions datas (dict)

5.9.2.2 `get_sum_concept_scores()`

```
def openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis.get_sum_concept_scores (
    self,
```



```

institutions,
concept_links )

```

Gets the sum of the concept scores of the concepts in the list concepts.

Parameters

<i>Institutions</i>	The institution (list of dict)
<i>concept_links</i>	The concept links

Returns

The sum of the concept scores

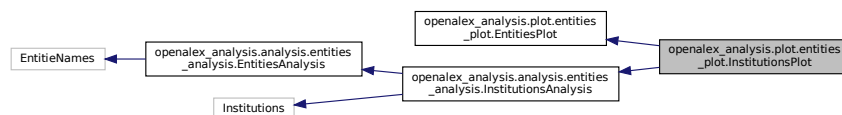
The documentation for this class was generated from the following file:

- openalex_analysis/analysis/entities_analysis.py

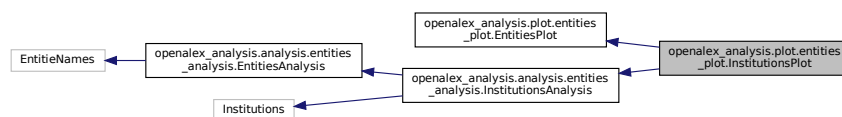
5.10 openalex_analysis.plot.entities_plot.InstitutionsPlot Class Reference

This class contains specific methods for Institutions concepts plot.

Inheritance diagram for openalex_analysis.plot.entities_plot.InstitutionsPlot:



Collaboration diagram for openalex_analysis.plot.entities_plot.InstitutionsPlot:



Public Member Functions

- def [getCustomData](#) (self, concept)
Gets the custom data for the plot.
- def [getHoverTemplate](#) (self, concept)
Gets the hover template for the plot.
- def [get_figure_institutions_multi_concepts_filtered](#) (self, plot_parameters, concepts_from, concepts_filters, thresholds, x_threshold, cited_by_threshold, institution_to_highlight)
Gets the figure with the institutions of multiple concepts and filtered.

Additional Inherited Members

5.10.1 Detailed Description

This class contains specific methods for Institutions concepts plot.

5.10.2 Member Function Documentation

5.10.2.1 `get_figure_institutions_multi_concepts_filtered()`

```
def openalex_analysis.plot.entities_plot.InstitutionsPlot.get_figure_institutions_multi_↔
concepts_filtered (
    self,
    plot_parameters,
    concepts_from,
    concepts_filters,
    thresholds,
    x_threshold,
    cited_by_threshold,
    institution_to_highlight )
```

Gets the figure with the institutions of multiple concepts and filtered.

Parameters

<i>plot_parameters</i>	The plot parameters (dict)
<i>concepts_from</i>	The concepts to import to create the dataset (list of str)
<i>concepts_filters</i>	The concepts to use to filter the institutions (list of str)
<i>thresholds</i>	The thresholds for each concept filter (list of float or int)
<i>x_threshold</i>	The global threshold (eg nb of works), usually corresponding to the x data (float or int)
<i>cited_by_threshold</i>	The cited by threshold (float or int)
<i>institution_to_highlight</i>	The institution to highlight on the plot (str)

Returns

The figure (fig)

5.10.2.2 `getCustomData()`

```
def openalex_analysis.plot.entities_plot.InstitutionsPlot.getCustomData (
    self,
    concept )
```

Gets the custom data for the plot.

Parameters

<i>concept</i>	The concept (str)
----------------	-------------------

Returns

The custom data (list of str)

5.10.2.3 getHoverTemplate()

```
def openalex_analysis.plot.entities_plot.InstitutionsPlot.getHoverTemplate (
    self,
    concept )
```

Gets the hover template for the plot.

Parameters

<i>concept</i>	The concept (str)
----------------	-------------------

Returns

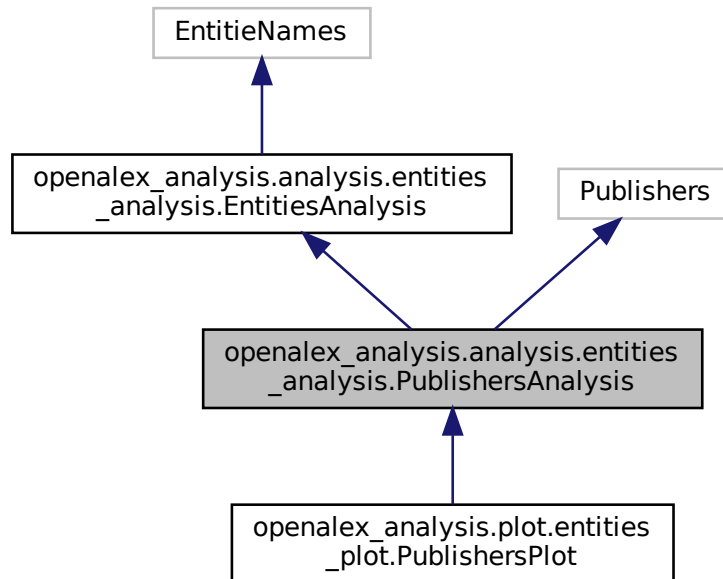
The hover template (list of str)

The documentation for this class was generated from the following file:

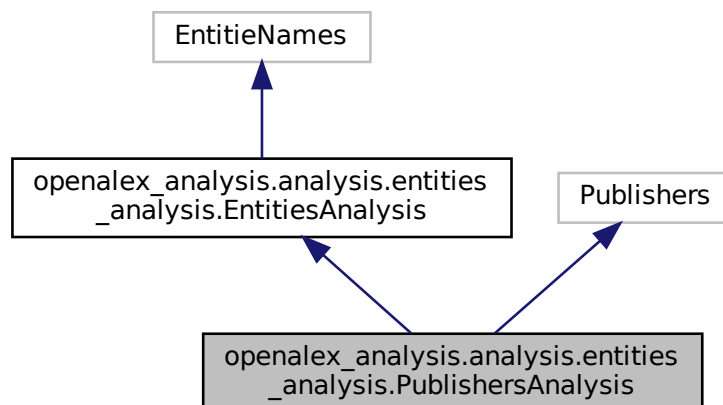
- openalex_analysis/plot/entities_plot.py

5.11 openalex_analysis.analysis.entities_analysis.PublishersAnalysis Class Reference

Inheritance diagram for openalex_analysis.analysis.entities_analysis.PublishersAnalysis:



Collaboration diagram for openalex_analysis.analysis.entities_analysis.PublishersAnalysis:



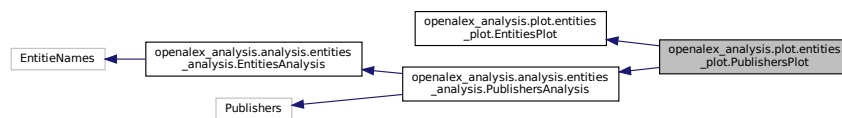
Additional Inherited Members

The documentation for this class was generated from the following file:

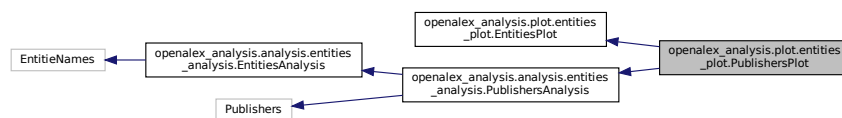
- openalex_analysis/analysis/entities_analysis.py

5.12 openalex_analysis.plot.entities_plot.PublishersPlot Class Reference

Inheritance diagram for openalex_analysis.plot.entities_plot.PublishersPlot:



Collaboration diagram for openalex_analysis.plot.entities_plot.PublishersPlot:



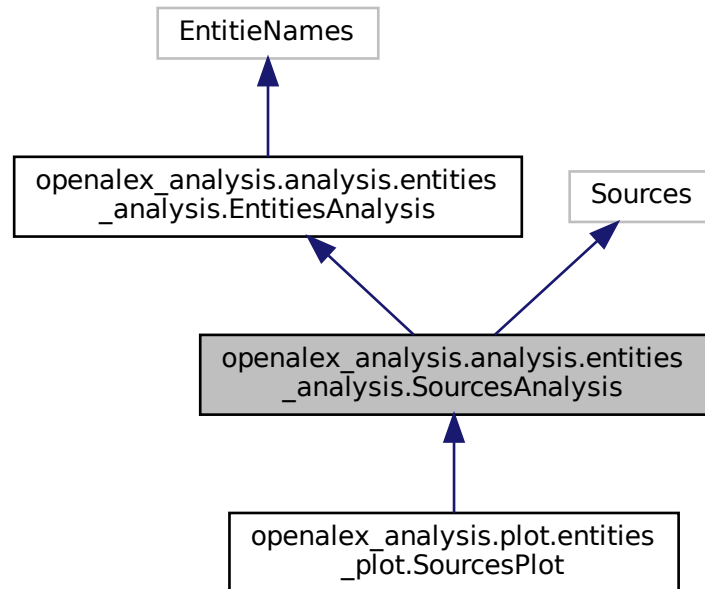
Additional Inherited Members

The documentation for this class was generated from the following file:

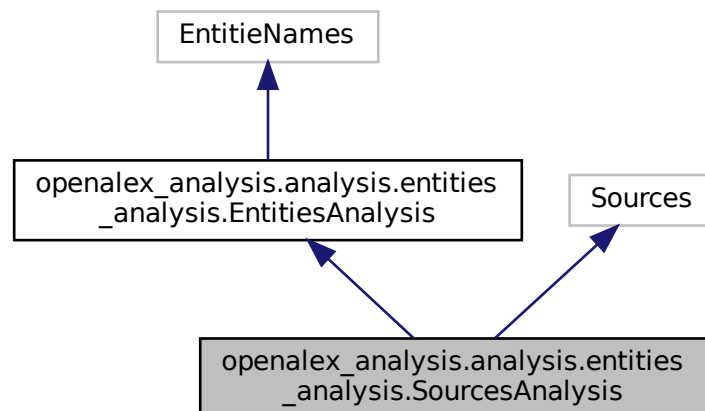
- openalex_analysis/plot/entities_plot.py

5.13 openalex_analysis.analysis.entities_analysis.SourcesAnalysis Class Reference

Inheritance diagram for openalex_analysis.analysis.entities_analysis.SourcesAnalysis:



Collaboration diagram for openalex_analysis.analysis.entities_analysis.SourcesAnalysis:



Static Public Attributes

- **EntitieOpenAlex** = Sources

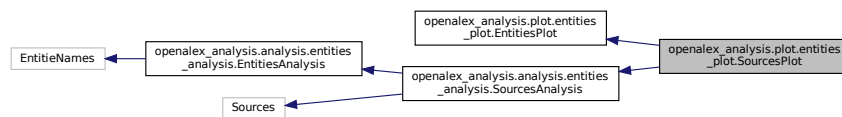
Additional Inherited Members

The documentation for this class was generated from the following file:

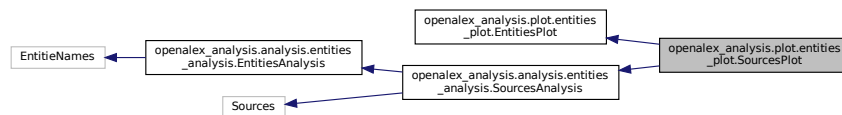
- openalex_analysis/analysis/entities_analysis.py

5.14 openalex_analysis.plot.entities_plot.SourcesPlot Class Reference

Inheritance diagram for openalex_analysis.plot.entities_plot.SourcesPlot:



Collaboration diagram for openalex_analysis.plot.entities_plot.SourcesPlot:



Additional Inherited Members

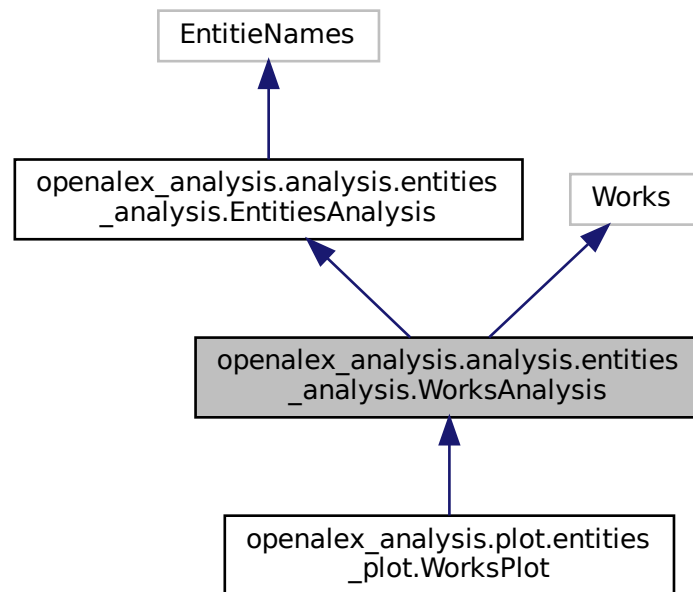
The documentation for this class was generated from the following file:

- openalex_analysis/plot/entities_plot.py

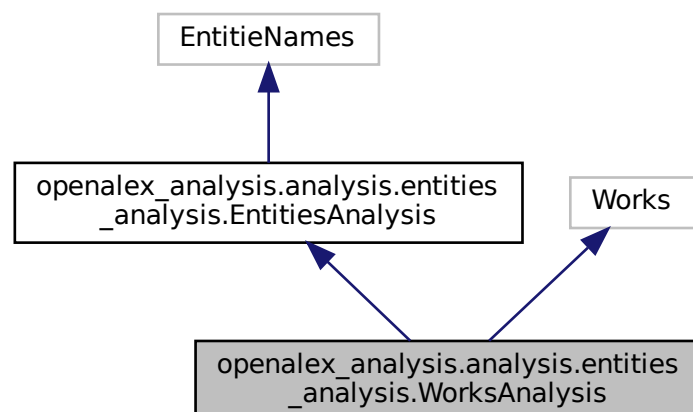
5.15 openalex_analysis.analysis.entities_analysis.WorksAnalysis Class Reference

This class contains specific methods for Works concepts analysis.

Inheritance diagram for openalex_analysis.analysis.entities_analysis.WorksAnalysis:



Collaboration diagram for openalex_analysis.analysis.entities_analysis.WorksAnalysis:



Public Member Functions

- def `filter_and_format_entitie_data_from_api_response` (self, entitie)

Filter and format the works data downloaded from the API.
- def `get_country_code` (self, entitie)

Gets the country code from an entitie.
- def `get_institution_name` (self, entitie)

Gets the institution name from an entitie.
- def `get_works_references_count` (self, count_years=[])

Gets the works references count of the works list of the instance.
- def `get_works_concepts_count` (self, count_years=[])

Gets the concepts count of the works list of the instance.
- def `get_element_count` (self, element_type, count_years=[])
- def `create_element_used_count_array` (self, element_type, entities_from=[], out_file_name=None, save_out_array=False, count_years=[])

Creates the element used count array.
- def `sort_count_array` (self, sort_by='h_used_all_l_use_main', sort_by_ascending=False)

Sort the dataframe with the count array (element_count_df)
- def `add_statistics_to_element_count_array` (self, sort_by='h_used_all_l_use_main', sort_by_ascending=False, min_concept_level=None)

Adds a statistics to the element count array (statistics between the main entitie to compare (second column in the dataframe) and the sum of the other entities)
- def `add_statistics_to_references_works_count_array` (self)

Adds a statistics to the references works count array (statistics between the main entitie to compare (second column in the dataframe) and the sum of the other entities)
- def `add_statistics_to_concept_count_array` (self, min_concept_level=None)

Adds a statistics to the concepts count array (statistics between the main entitie to compare (second column in the dataframe) and the sum of the other entities)

Public Attributes

- `entitie_from_type`
- `count_element_type`
- `count_element_years`
- `count_entities_cols`
- `entitie_from_id`
- `element_count_df`
- `create_element_count_array_progress_percentage`
- `create_element_count_array_progress_text`

Static Public Attributes

- `EntitieOpenAlex` = Works

5.15.1 Detailed Description

This class contains specific methods for Works concepts analysis.

5.15.2 Member Function Documentation

5.15.2.1 add_statistics_to_element_count_array()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.add_statistics_to_element_↵
count_array (
    self,
    sort_by = 'h_used_all_l_use_main',
    sort_by_ascending = False,
    min_concept_level = None )
```

Adds a statistics to the element count array (statistics between the main entitie to compare (second column in the dataframe) and the sum of the other entities)

Parameters

<i>sort_by</i>	The key to sort the dataframe (str)
<i>sort_by_ascending</i>	Whenever to sort the dataframe ascending (bool)
<i>min_concept_level</i>	In case the element is a concept, this is the minimum level of the concepts we will keep (aka remove the lower (= more global) concepts)

5.15.2.2 create_element_used_count_array()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.create_element_used_count_array
(
    self,
    element_type,
    entities_from = [],
    out_file_name = None,
    save_out_array = False,
    count_years = [] )
```

Creates the element used count array.

Count the number of times each element (eg reference, concept..) is used and save the array as CSV (optional)

Parameters

<i>element_type</i>	The element type
<i>entities_from</i>	The extra entities to which to count the concepts (list of str)
<i>out_file_name</i>	The out CSV file name, if not provided, an appropriate name is generated (str)
<i>save_out_array</i>	Save out array (bool)
<i>count_years</i>	If given, it will compute the count for each year (list[int])

5.15.2.3 filter_and_format_entitie_data_from_api_response()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.filter_and_format_entitie_data_from_api_response (
    self,
    entitie )
```

Filter and format the works data downloaded from the API.

Parameters

<i>entitie</i>	The works data from the API (dict)
----------------	------------------------------------

Returns

The works datas (dict)

5.15.2.4 get_country_code()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.get_country_code (
    self,
    entitie )
```

Gets the country code from an entitie.

Parameters

<i>entitie</i>	The entitie (dict)
----------------	--------------------

Returns

The country code (str)

5.15.2.5 get_institution_name()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.get_institution_name (
    self,
    entitie )
```

Gets the institution name from an entitie.

Parameters

<i>entitie</i>	The entitie (dict)
----------------	--------------------

Returns

The institution name (str)

5.15.2.6 get_works_concepts_count()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.get_works_concepts_count (
    self,
    count_years = [] )
```

Gets the concepts count of the works list of the instance.

Parameters

<i>count_only_year</i>	If different than None, count only the concepts of the works of the given year (int)
------------------------	--

Returns

The concept count (pandas Serie)

5.15.2.7 get_works_references_count()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.get_works_references_count (
    self,
    count_years = [] )
```

Gets the works references count of the works list of the instance.

Returns

The works references count (pandas Serie)

5.15.2.8 sort_count_array()

```
def openalex_analysis.analysis.entities_analysis.WorksAnalysis.sort_count_array (
    self,
    sort_by = 'h_used_all_l_use_main',
    sort_by_ascending = False )
```

Sort the dataframe with the count array (element_count_df)

Parameters

<i>sort_by</i>	The key to sort the dataframe (str)
<i>sort_by_ascending</i>	Whenever to sort the dataframe ascending (bool)

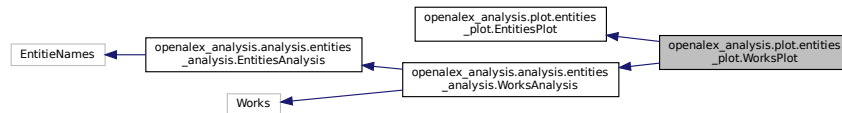
The documentation for this class was generated from the following file:

- openalex_analysis/analysis/entities_analysis.py

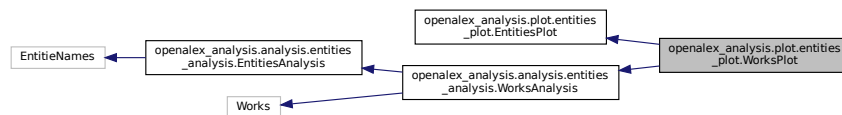
5.16 openalex_analysis.plot.entities_plot.WorksPlot Class Reference

This class contains specific methods for Works concepts plot.

Inheritance diagram for openalex_analysis.plot.entities_plot.WorksPlot:



Collaboration diagram for openalex_analysis.plot.entities_plot.WorksPlot:



Public Member Functions

- def [getCustomData](#) (self, concept)
Gets the custom data for the plot.
- def [getHoverTemplate](#) (self, concept)
Gets the hover template for the plot.
- def [get_figure_nb_time_referenced](#) (self, element_type)
Gets the figure with the number of time each reference is used in a list of works.

Additional Inherited Members

5.16.1 Detailed Description

This class contains specific methods for Works concepts plot.

5.16.2 Member Function Documentation

5.16.2.1 get_figure_nb_time_referenced()

```
def openalex_analysis.plot.entities_plot.WorksPlot.get_figure_nb_time_referenced (
    self,
    element_type )
```

Gets the figure with the number of time each reference is used in a list of works.

Parameters

<i>element_type</i>	The element type
---------------------	------------------

Returns

The figure works number of time referenced (fig)

5.16.2.2 getCustomData()

```
def openalex_analysis.plot.entities_plot.WorksPlot.getCustomData (
    self,
    concept )
```

Gets the custom data for the plot.

Parameters

<i>concept</i>	The concept (str)
----------------	-------------------

Returns

The custom data (list of str)

5.16.2.3 getHoverTemplate()

```
def openalex_analysis.plot.entities_plot.WorksPlot.getHoverTemplate (
    self,
    concept )
```

Gets the hover template for the plot.

Parameters

<i>concept</i>	The concept (str)
----------------	-------------------

Returns

The hover template (list of str)

The documentation for this class was generated from the following file:

- openalex_analysis/plot/entities_plot.py

Index

<code>__init__</code>	<code>get_figure_entities_selection_threshold</code>
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 19	<code>openalex_analysis.plot.entities_plot.EntitiesPlot</code> , 25
<code>add_average_combined_concept_score_to_multi_concept_entity_df</code>	<code>get_figure_institutions_multi_concepts_filtered</code>
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 20	<code>openalex_analysis.plot.entities_plot.InstitutionsPlot</code> , 30
<code>add_statistics_to_element_count_array</code>	<code>get_figure_nb_time_referenced</code>
<code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> , 38	<code>openalex_analysis.plot.entities_plot.WorksPlot</code> , 41
<code>auto_remove_databases_saved</code>	<code>get_figure_time_series_element_used_by_entities</code>
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 20	<code>openalex_analysis.plot.entities_plot.EntitiesPlot</code> , 26
<code>create_element_used_count_array</code>	<code>get_institution_name</code>
<code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> , 38	<code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> , 39
<code>create_multi_concept_filters_entities_dataframe</code>	<code>get_name_of_entity</code>
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 20	<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 23
<code>filter_and_format_entity_data_from_api_response</code>	<code>get_number_of_entities_selected</code>
<code>openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis</code> , 28	<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 23
<code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> , 38	<code>get_sum_concept_scores</code>
<code>get_api_query</code>	<code>openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis</code> , 28
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 21	<code>get_works_concepts_count</code>
<code>get_count_entities_matched</code>	<code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> , 40
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 21	<code>get_works_references_count</code>
<code>get_country_code</code>	<code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> , 40
<code>openalex_analysis.analysis.entities_analysis.WorksAnalysis</code> , 39	<code>getCustomData</code>
<code>get_database_file_name</code>	<code>openalex_analysis.plot.entities_plot.InstitutionsPlot</code> , 30
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 21	<code>openalex_analysis.plot.entities_plot.WorksPlot</code> , 42
<code>get_df_filtered_entities_selection_threshold</code>	<code>getInstitutionTemplate</code>
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 22	<code>openalex_analysis.plot.entities_plot.InstitutionsPlot</code> , 31
<code>get_entity_string_name</code>	<code>openalex_analysis.plot.entities_plot.WorksPlot</code> , 42
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 22	<code>openalex_analysis.analysis.entities_analysis.AnalysisConfig</code> , 11
<code>get_entity_type_from_id</code>	<code>openalex_analysis.analysis.entities_analysis.AuthorsAnalysis</code> , 13
<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 22	<code>openalex_analysis.analysis.entities_analysis.ConceptsAnalysis</code> , 15
<code>get_figure_entities_of_a_concept_color_country</code>	<code>openalex_analysis.analysis.entities_analysis.EntitiesAnalysis</code> , 17
<code>openalex_analysis.plot.entities_plot.EntitiesPlot</code> , 25	<code>__init__</code> , 19
	<code>add_average_combined_concept_score_to_multi_concept_entity_df</code>
	<code>auto_remove_databases_saved</code> , 20

- create_multi_concept_filters_entities_dataframe, 20
- get_api_query, 21
- get_count_entities_matched, 21
- get_database_file_name, 21
- get_df_filtered_entities_selection_threshold, 22
- get_entitie_string_name, 22
- get_entitie_type_from_id, 22
- get_name_of_entitie, 23
- get_number_of_entities_selected, 23
- openalex_analysis.analysis.entities_analysis.InstitutionsAnalysis, 27
 - filter_and_format_entitie_data_from_api_response, 28
 - get_sum_concept_scores, 28
- openalex_analysis.analysis.entities_analysis.PublishersAnalysis, 32
- openalex_analysis.analysis.entities_analysis.SourcesAnalysis, 34
- openalex_analysis.analysis.entities_analysis.WorksAnalysis, 36
 - add_statistics_to_element_count_array, 38
 - create_element_used_count_array, 38
 - filter_and_format_entitie_data_from_api_response, 38
 - get_country_code, 39
 - get_institution_name, 39
 - get_works_concepts_count, 40
 - get_works_references_count, 40
 - sort_count_array, 40
- openalex_analysis.names.entitie_names.EntitieNames, 17
- openalex_analysis.plot.entities_plot.AuthorsPlot, 14
- openalex_analysis.plot.entities_plot.ConceptsPlot, 16
- openalex_analysis.plot.entities_plot.EntitiesPlot, 24
 - get_figure_entities_of_a_concept_color_country, 25
 - get_figure_entities_selection_threshold, 25
 - get_figure_time_series_element_used_by_entities, 26
- openalex_analysis.plot.entities_plot.InstitutionsPlot, 29
 - get_figure_institutions_multi_concepts_filtered, 30
 - getCustomData, 30
 - getHoverTemplate, 31
- openalex_analysis.plot.entities_plot.PublishersPlot, 33
- openalex_analysis.plot.entities_plot.SourcesPlot, 35
- openalex_analysis.plot.entities_plot.WorksPlot, 41
 - get_figure_nb_time_referenced, 41
 - getCustomData, 42
 - getHoverTemplate, 42
- sort_count_array
 - openalex_analysis.analysis.entities_analysis.WorksAnalysis, 40