Lớp xây dựng

Buildersdatasets.DatasetBuilder

Bộ dữ liệu dựa trên hai lớp chính trong quá trình xây dựng tập dữ liệu: DatasetBuilder và BuilderConfig.

bộ dữ liệu lớp.DatasetBuilderdatasets.DatasetBuilderhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L210[{"name": "cache_dir", "val": ": typing.Optional[str] = None"}, {"name": "dataset_name", "val": ": typing.Optional[str] = None"}, {"name": "config_name", "val": ": typing.Optional[str] = None"}, {"name": "hash", "val": ": typing.Optional[str] = None"}, {"name": "base_path", "val": ": typing.Optional[str] = None"}, {"name": "info", "val": ": typing.Optional[datasets.info.DatasetInfo] = None"}, {"name": "features", "val": ": typing.Optional[datasets.features.features.Features] = None"}, {"name": "token", "val": ": typing.Union[bool, str, NoneType] = None"}, {"name": "repo_id", "val": ": typing.Optional[str] = None"}, {"name": "data_files", "val": ": typing.Union[str, list, dict, datasets.data_files.DataFilesDict, NoneType] = None"}, {"name": "data_dir", "val": ": typing.Optional[str] = None"}, {"name": "storage_options", "val": ": typing.Optional[dict] = None"}, {"name": "writer_batch_size", "val": ": typing.Optional[int] = None"}, {"name": "**config_kwargs", "val": ""}]- cache_dir (str , optional) --
Thư mục để lưu trữ dữ liệu. Mặc định là "~/.cache/huggingface/datasets" .

- dataset_name (str , optional) --
  Tên của tập dữ liệu, nếu khác với tên trình tạo. Hữu ích cho các nhà xây dựng đóng gói như csv, thư mục hình ảnh, thư mục âm thanh, v.v. để phản ánh sự khác biệt giữa các tập dữ liệu sử dụng cùng một trình xây dựng đóng gói.
- config_name (str , optional) --
  Tên của cấu hình tập dữ liệu.
  Nó ảnh hưởng đến dữ liệu được tạo trên đĩa. Các cấu hình khác nhau sẽ có cái riêng thư mục con và
  các phiên bản.
  If not provided, the default configuration is used (if it exists).
  Tên tham số đã được đổi tên thành config_name .

- hash (str , optional) --

Băm cụ thể cho mã xây dựng tập dữ liệu. Được sử dụng để cập nhật thư mục bộ đệm khi dataset builder code is updated (to avoid reusing old data).
The typical caching directory (defined in self._relative_data_dir) is tên/phiên bản/băm/ .

- base_path (str , optional) --
  Đường dẫn cơ sở cho các đường dẫn tương đối được sử dụng để tải xuống tệp.
  Đây có thể là một URL từ xa.
- features (Features, optional) --
  Các loại tính năng sẽ sử dụng với tập dữ liệu này.
  Ví dụ, nó có thể được sử dụng để thay đổi các loại Tính năng của tập dữ liệu.
- token (str or bool , optional) --
  Chuỗi hoặc boolean để sử dụng làm mã thông báo Bearer cho các tệp từ xa trên Trung tâm bộ dữ liệu. Nếu True , sẽ nhận được token từ "~/.huggingface" .
- repo_id (str , optional) --
  ID của kho lưu trữ dữ liệu.
  Dùng để phân biệt các nhà xây dựng có cùng tên nhưng không cùng nguồn gốc không gian tên, ví dụ "rajpurkar/squad"
  và ID repo "lhoestq/squad". Trong trường hợp sau, tên của người xây dựng sẽ là "lhoestq___squad".
- data_files (str or Sequence or Mapping , optional) --
  Path(s) to source data file(s).
  Dành cho các trình tạo như "csv" hoặc "json" cần người dùng chỉ định tệp dữ liệu. Họ có thể là một trong
  tập tin cục bộ hoặc từ xa. Để thuận tiện, bạn có thể sử dụng DataFilesDict .
- data_dir (str , optional) --
  Path to directory containing source data file(s).
  Chỉ sử dụng nếu data_files không được truyền, trong trường hợp đó nó tương đương với việc truyền os.path.join(data_dir, "**") as data_files .
  Đối với các nhà xây dựng yêu cầu tải xuống thủ công, đó phải là đường dẫn đến thư mục cục bộ chứa
  dữ liệu được tải xuống theo cách thủ công.
- storage_options (dict , optional) --
  Các cặp khóa/giá trị sẽ được chuyển đến phần phụ trợ hệ thống tệp tập dữ liệu, nếu có.
- writer_batch_size (int , optional) --
  Kích thước lô được ArrowWriter sử dụng.
  Nó xác định số lượng mẫu được lưu trong bộ nhớ trước khi ghi chúng
  và cả chiều dài của các đoạn mũi tên.

Không có nghĩa là ArrowWriter sẽ sử dụng giá trị mặc định của nó.

- **config_kwargs (additional keyword arguments) -- Keyword arguments to be passed to người xây dựng tương ứng
  lớp cấu hình, được đặt trên thuộc tính lớp DatasetBuilder.BUILDER_CONFIG_CLASS. Người xây dựng
  lớp cấu hình là BuilderConfig hoặc một lớp con của nó.0
  Lớp cơ sở trừu tượng cho tất cả các tập dữ liệu.

DatasetBuilder có 3 phương thức chính:

- DatasetBuilder.info : Ghi lại tập dữ liệu, bao gồm cả tính năng
  tên, loại, hình dạng, phiên bản, phần tách, trích dẫn, v.v.
- DatasetBuilder.download_and_prepare(): Downloads the source data
  và ghi nó vào đĩa.
- DatasetBuilder.as_dataset(): Generates a Dataset.

Một số DatasetBuilder hiển thị nhiều biến thể của
tập dữ liệu bằng cách định nghĩa một lớp con BuilderConfig và chấp nhận một
config object (or name) on construction. Configurable datasets expose a
pre-defined set of configurations in DatasetBuilder.builder_configs() .

as_datasetdatasets.DatasetBuilder.as_datasethttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L1027[{"name": "split", "val": ": typing.Union[str, datasets.splits.Split, list[str], list[datasets.splits.Split], NoneType] = None"}, {"name": "run_post_process", "val": " = True"}, {"name": "verification_mode", "val": ": typing.Union[datasets.utils.info_utils.VerificationMode, str, NoneType] = None"}, {"name": "in_memory", "val": " = False"}]- split (datasets.Split) --
Tập hợp con nào của dữ liệu sẽ được trả về.

- run_post_process (bool , defaults to True) --
  Có chạy các phép biến đổi và/hoặc thêm tập dữ liệu xử lý hậu kỳ hay không
  chỉ mục.
- verification_mode (VerificationMode or str , defaults to BASIC_CHECKS) --
  Chế độ xác minh xác định các hoạt động kiểm tra sẽ chạy trên
  downloaded/processed dataset information (checksums/size/splits/...).
- in_memory (bool , defaults to False) --
  Có sao chép dữ liệu trong-memory.0datasets.Dataset hay không

Trả về Tập dữ liệu cho phần tách được chỉ định.

Ví dụ:

```
>>> from datasets import load_dataset_builder
>>> builder = load_dataset_builder('cornell-movie-review-data/rotten_tomatoes')
>>> builder.download_and_prepare()
>>> ds = builder.as_dataset(split='train')
>>> ds
Dataset({
    features: ['text', 'label'],
    num_rows: 8530
})
```

download_and_preparedatasets.DatasetBuilder.download_and_preparehttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L691[{"name": "output_dir", "val": ": typing.Optional[str] = None"}, {"name": "download_config", "val": ": typing.Optional[datasets.download.download_config.DownloadConfig] = None"}, {"name": "download_mode", "val": ": typing.Union[datasets.download.download_manager.DownloadMode, str, NoneType] = None"}, {"name": "verification_mode", "val": ": typing.Union[datasets.utils.info_utils.VerificationMode, str, NoneType] = None"}, {"name": "dl_manager", "val": ": typing.Optional[datasets.download.download_manager.DownloadManager] = None"}, {"name": "base_path", "val": ": typing.Optional[str] = None"}, {"name": "file_format", "val": ": str = 'arrow'"}, {"name": "max_shard_size", "val": ": typing.Union[str, int, NoneType] = None"}, {"name": "num_proc", "val": ": typing.Optional[int] = None"}, {"name": "storage_options", "val": ": typing.Optional[dict] = None"}, {"name": "**download_and_prepare_kwargs", "val": ""}]- output_dir (str , optional) --
Thư mục đầu ra cho tập dữ liệu.
Mặc định là cache_dir của trình tạo này, nằm trong ~/.cache/huggingface/datasets theo mặc định.

- **download_config** (`DownloadConfig`, *optional*) -- Specific download configuration parameters. - **download_mode** ([DownloadMode](/docs/datasets/v4.2.0/en/package_reference/builder_classes#datasets.DownloadMode) or `str`, *optional*) -- Select the chế độ tải xuống/tạo, mặc định là `REUSE_DATASET_IF_EXISTS`. - **chế độ xác minh** ([VerificationMode](/docs/datasets/v4.2.0/en/package_reference/builder_classes#datasets.VerificationMode) or `str`, defaults to `BASIC_CHECKS`) --

Chế độ xác minh xác định các bước kiểm tra để chạy trên tập dữ liệu đã tải xuống/đã xử lý information (checksums/size/splits/...). - **dl_manager** (`DownloadManager`, *optional*) -- Specific `DownloadManger` to use. - **base_path** (`str`, *optional*) -- Base path for relative đường dẫn được sử dụng để tải tập tin. Đây có thể là một url từ xa. Nếu không được chỉ định, giá trị của the `base_path` attribute (`self.base_path`) will be used instead. - **file_format** (`str`, *optional*) -- Format of the data files in which the dataset will be written. Supported formats: "mũi tên", "sàn gỗ". Mặc định là định dạng "mũi tên". Nếu định dạng là "sàn gỗ", thì hình ảnh và âm thanh dữ liệu được nhúng vào tệp Parquet thay vì trỏ đến tệp cục bộ. - **max_shard_size** (`Union[str, int]`, *optional*) -- Maximum number of bytes written per shard, default is "500MB". Kích thước dựa trên kích thước dữ liệu không nén, vì vậy trong thực tế, tệp phân đoạn của bạn có thể nhỏ hơ than `max_shard_size` thanks to Parquet compression for example. - **num_proc** (`int`, *optional*, defaults to `None`) -- Number of processes when downloading and generating the dataset locally. Multiprocessing is disabled by default. - **storage_options** (`dict`, *optional*) -- Các cặp khóa/giá trị sẽ được chuyển đến phần phụ trợ hệ thống tệp bộ nhớ đệm, nếu có. - ****download_and_prepare_kwargs** (additional keyword arguments) -- Keyword arguments.0
Tải xuống và chuẩn bị dữ liệu để đọc.
Ví dụ:

Tải xuống và chuẩn bị tập dữ liệu dưới dạng tệp Mũi tên có thể được tải dưới dạng Tập dữ liệu bằng cách sử builder.as_dataset() :

```
>>> from datasets import load_dataset_builder
>>> builder = load_dataset_builder("cornell-movie-review-data/rotten_tomatoes")
>>> builder.download_and_prepare()
```

Tải xuống và chuẩn bị tập dữ liệu dưới dạng tệp Parquet được phân chia cục bộ:

```
>>> from datasets import load_dataset_builder
>>> builder = load_dataset_builder("cornell-movie-review-data/rotten_tomatoes")
>>> builder.download_and_prepare("./output_dir", file_format="parquet")
```

Tải xuống và chuẩn bị tập dữ liệu dưới dạng tệp Parquet được phân chia trong bộ lưu trữ đám mây:

```
>>> from datasets import load_dataset_builder
>>> storage_options = {"key": aws_access_key_id, "secret": aws_secret_access_key}
>>> builder = load_dataset_builder("cornell-movie-review-data/rotten_tomatoes")
>>> builder.download_and_prepare("s3://my-bucket/my_rotten_tomatoes", storage_options=storage_opti
```

get_imported_module_dirdatasets.DatasetBuilder.get_imported_module_dirhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L683[]
Trả về đường dẫn của mô-đun của lớp hoặc lớp con này.

bộ dữ liệu lớp.GeneratorBasedBuilderdatasets.GeneratorBasedBuilderhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L1356[{"name": "cache_dir", "val": ": typing.Optional[str] = None"}, {"name": "dataset_name", "val": ": typing.Optional[str] = None"}, {"name": "config_name", "val": ": typing.Optional[str] = None"}, {"name": "hash", "val": ": typing.Optional[str] = None"}, {"name": "base_path", "val": ": typing.Optional[str] = None"}, {"name": "info", "val": ": typing.Optional[datasets.info.DatasetInfo] = None"}, {"name": "features", "val": ": typing.Optional[datasets.features.features.Features] = None"}, {"name": "token", "val": ": typing.Union[bool, str, NoneType] = None"}, {"name": "repo_id", "val": ": typing.Optional[str] = None"}, {"name": "data_files", "val": ": typing.Union[str, list, dict, datasets.data_files.DataFilesDict, NoneType] = None"}, {"name": "data_dir", "val": ": typing.Optional[str] = None"}, {"name": "storage_options", "val": ": typing.Optional[dict] = None"}, {"name": "writer_batch_size", "val": ": typing.Optional[int] = None"}, {"name": "**config_kwargs", "val": ""}]
Lớp cơ sở cho các bộ dữ liệu có khả năng tạo dữ liệu dựa trên trình tạo dict.

GeneratorBasedBuilder là một lớp tiện lợi trừu tượng hóa nhiều
về việc ghi và đọc dữ liệu của DatasetBuilder. Nó mong đợi các lớp con
triển khai trình tạo từ điển tính năng trên các phần tách tập dữ liệu
(_split_generators). See the method docstrings for details.

bộ dữ liệu lớp.ArrowBasedBuilderdatasets.ArrowBasedBuilderhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L1621[{"name": "cache_dir", "val": ": typing.Optional[str] = None"}, {"name": "dataset_name", "val": ": typing.Optional[str] = None"}, {"name": "config_name", "val": ": typing.Optional[str] = None"}, {"name": "hash", "val": ": typing.Optional[str] = None"}, {"name": "base_path", "val": ": typing.Optional[str] = None"}, {"name": "info", "val": ": typing.Optional[datasets.info.DatasetInfo] = None"}, {"name": "features", "val": ": typing.Optional[datasets.features.features.Features] = None"}, {"name":

"token", "val": ": typing.Union[bool, str, NoneType] = None"}, {"name": "repo_id", "val": ": typing.Optional[str] = None"}, {"name": "data_files", "val": ": typing.Union[str, list, dict, datasets.data_files.DataFilesDict, NoneType] = None"}, {"name": "data_dir", "val": ": typing.Optional[str] = None"}, {"name": "storage_options", "val": ": typing.Optional[dict] = None"}, {"name": "writer_batch_size", "val": ": typing.Optional[int] = None"}, {"name": "**config_kwargs", "val": ""}]
Base class for datasets with data generation based on Arrow loading functions (CSV/JSON/Parquet).

bộ dữ liệu lớp.BuilderConfigdatasets.BuilderConfighttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L97[{"name": "name", "val": ": str = 'default'"}, {"name": "version", "val": ": typing.Union[datasets.utils.version.Version, str, NoneType] = 0.0.0"}, {"name": "data_dir", "val": ": typing.Optional[str] = None"}, {"name": "data_files", "val": ": typing.Union[datasets.data_files.DataFilesDict, datasets.data_files.DataFilesPatternsDict, NoneType] = None"}, {"name": "description", "val": ": typing.Optional[str] = None"}]- name (str , defaults to default) --
Tên của cấu hình.

- version (Version or str , defaults to 0.0.0) --
  Phiên bản của cấu hình.
- data_dir (str , optional) --
  Đường dẫn đến thư mục chứa dữ liệu nguồn.
- data_files (str or Sequence or Mapping , optional) --
  Path(s) to source data file(s).
- description (str , optional) --
  Mô tả của con người về cấu hình.0
  Lớp cơ sở cho cấu hình dữ liệu DatasetBuilder.

Các lớp con DatasetBuilder với các tùy chọn cấu hình dữ liệu sẽ được phân lớp BuilderConfig và thêm thuộc tính của riêng mình.

create_config_iddatasets.BuilderConfig.create_config_idhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/builder.py#L140[{"name": "config_kwargs", "val": ": dict"}, {"name": "custom_features", "val": ": typing.Optional[datasets.features.features.Features] = None"}]

Id cấu hình được sử dụng để xây dựng thư mục bộ đệm.

Theo mặc định, nó bằng tên cấu hình.
Tuy nhiên, tên của cấu hình là không đủ để có mã định danh duy nhất cho tập dữ liệu đang được tạo ra
vì nó không tính đến:

- kwargs cấu hình có thể được sử dụng để ghi đè các thuộc tính
- các tính năng tùy chỉnh được sử dụng để ghi tập dữ liệu
- data_files cho bộ dữ liệu json/text/csv/pandas

Do đó, id cấu hình chỉ là tên cấu hình có hậu tố tùy chọn dựa trên các tên này.

Downloaddatasets.DownloadManager

tập dữ liệu lớp.DownloadManagerdatasets.DownloadManagerhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/download_manager.py#L71[{"name": "dataset_name", "val": ": typing.Optional[str] = None"}, {"name": "data_dir", "val": ": typing.Optional[str] = None"}, {"name": "download_config", "val": ": typing.Optional[datasets.download.download_config.DownloadConfig] = None"}, {"name": "base_path", "val": ": typing.Optional[str] = None"}, {"name": "record_checksums", "val": " = True"}]

downloaddatasets.DownloadManager.downloadhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/download_manager.py#L131[{"name": "url_or_urls", "val": ""}]-
url_or_urls (str or list or dict) --
URL hoặc danh sách hoặc mệnh lệnh của URL để tải xuống. Mỗi URL là một str .0 str hoặc danh sách hoặc di
đường dẫn đã tải xuống khớp với url_or_urls đầu vào đã cho.
Download given URL(s).

Theo mặc định, chỉ có một quy trình được sử dụng để tải xuống. Vượt qua tùy chỉnh download_config.num_proc để thay đổi hành vi này.

Ví dụ:

```
>>> downloaded_files = dl_manager.download('https://storage.googleapis.com/seldon-datasets/sentenc
```

download_and_extractdatasets.DownloadManager.download_and_extracthttps://github.com/

huggingface/datasets/blob/4.2.0/src/datasets/download/download_manager.py#L310[{"name": "url_or_urls", "val": ""}]- url_or_urls (str or list or dict) --
URL hoặc danh sách hoặc chính tả các URL để tải xuống và giải nén. Mỗi URL là một
str .0extracted_path(s) str , extracted paths of given URL(s).
Tải xuống và giải nén url_or_urls đã cho.

Gần tương đương với:

```
extracted_paths = dl_manager.extract(dl_manager.download(url_or_urls))
```

extractdatasets.DownloadManager.extracthttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/download_manager.py#L278[{"name": "path_or_paths", "val": ""}]- path_or_paths (path or list or dict) --
Path of file to extract. Each path is a str .0extracted_path(s) str , The extracted paths
phù hợp với đầu vào nhất định
path_or_paths.
Extract given path(s).

Ví dụ:

```
>>> downloaded_files = dl_manager.download('https://storage.googleapis.com/seldon-datasets/sentenc
>>> extracted_files = dl_manager.extract(downloaded_files)
```

iter_archivedatasets.DownloadManager.iter_archivehttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/download_manager.py#L234[{"name": "path_or_buf", "val": ": typing.Union[str, _io.BufferedReader]"}]- path_or_buf (str or io.BufferedReader) --
Archive path or archive binary file object.0 tuple[str, io.BufferedReader] 2-tuple
(path_within_archive, file_object).
Đối tượng tệp được mở ở chế độ nhị phân.
Lặp lại các tập tin trong một kho lưu trữ.

Ví dụ:

```
>>> archive = dl_manager.download('https://storage.googleapis.com/seldon-datasets/sentence_polarit
>>> files = dl_manager.iter_archive(archive)
```

iter_filesdatasets.DownloadManager.iter_fileshttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/download_manager.py#L259[{"name": "paths", "val": ": typing.Union[str, list[str]]"}]- paths (str or list of str) --
Root path.0 str Đường dẫn tệp.
Lặp lại các đường dẫn tập tin.

Ví dụ:

```
>>> files = dl_manager.download_and_extract('https://huggingface.co/datasets/beans/resolve/main/da
>>> files = dl_manager.iter_files(files)
```

lớp học
bộ dữ liệu.StreamingDownloadManagerdatasets.StreamingDownloadManagerhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/streaming_download_manager.py#L47[{"name": "dataset_name", "val": ": typing.Optional[str] = None"}, {"name": "data_dir", "val": ": typing.Optional[str] = None"}, {"name": "download_config", "val": ": typing.Optional[datasets.download.download_config.DownloadConfig] = None"}, {"name": "base_path", "val": ": typing.Optional[str] = None"}]

Download manager that uses the "::" separator to navigate through (possibly remote)
kho lưu trữ nén.
Ngược lại với DownloadManager thông thường, các phương thức tải xuống và giải nén không thực sự tải xuống cũng không giải nén
dữ liệu, nhưng chúng trả về đường dẫn hoặc url có thể mở được bằng hàm xopen
mở rộng
chức năng mở tích hợp để truyền dữ liệu từ các tập tin từ xa.

downloaddatasets.StreamingDownloadManager.downloadhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/streaming_download_manager.py#L75[{"name": "url_or_urls", "val": ""}]- url_or_urls (str or list or dict) --
URL(s) of files to stream data from. Each url is a str .0url(s)(str or list or dict), URL(s)
để truyền dữ liệu từ việc khớp với url_or_urls đầu vào đã cho.
Normalize URL(s) of files to stream data from.
Đây là phiên bản lười biếng của DownloadManager.download để phát trực tuyến.

Ví dụ:

```
>>> downloaded_files = dl_manager.download('https://storage.googleapis.com/seldon-datasets/sentenc
```

download_and_extractdatasets.StreamingDownloadManager.download_and_extracthttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/streaming_download_manager.py#L151[{"name": "url_or_urls", "val": ""}]- url_or_urls (str or list or dict) --
URL(s) to stream from data from. Each url is a str .0url(s)(str or list or dict), URL(s) to truyền dữ liệu từ việc khớp với đầu vào đã cho url_or_urls .
Prepare given url_or_urls for streaming (add extraction protocol).

Đây là phiên bản lười biếng của DownloadManager.download_and_extract để phát trực tuyến.

Tương đương với:

```
urls = dl_manager.extract(dl_manager.download(url_or_urls))
```

extractdatasets.StreamingDownloadManager.extracthttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/streaming_download_manager.py#L102[{"name": "url_or_urls", "val": ""}]- url_or_urls (str or list or dict) --
URL(s) of files to stream data from. Each url is a str .0url(s)(str or list or dict), URL(s) để truyền dữ liệu từ việc khớp với đầu vào đã cho url_or_urls .
Add extraction protocol for given url(s) for streaming.

Đây là phiên bản lười biếng của DownloadManager.extract để phát trực tuyến.

Ví dụ:

```
>>> downloaded_files = dl_manager.download('https://storage.googleapis.com/seldon-datasets/sentenc
>>> extracted_files = dl_manager.extract(downloaded_files)
```

iter_archivedatasets.StreamingDownloadManager.iter_archivehttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/streaming_download_manager.py#L171[{"name": "urlpath_or_buf", "val": ": typing.Union[str, _io.BufferedReader]"}]- urlpath_or_buf (str or io.BufferedReader) --
Archive path or archive binary file object.0 tuple[str, io.BufferedReader] 2-tuple (path_within_archive, file_object).

Đối tượng tệp được mở ở chế độ nhị phân.
Lặp lại các tập tin trong một kho lưu trữ.

Ví dụ:

```
>>> archive = dl_manager.download('https://storage.googleapis.com/seldon-datasets/sentence_polarit
>>> files = dl_manager.iter_archive(archive)
```

iter_filesdatasets.StreamingDownloadManager.iter_fileshttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/streaming_download_manager.py#L196[{"name": "urlpaths", "val": ": typing.Union[str, list[str]]"}]- urlpaths (str or list of str) --
Đường dẫn URL gốc path.0strFile.
Lặp lại các tập tin.

Ví dụ:

```
>>> files = dl_manager.download_and_extract('https://huggingface.co/datasets/beans/resolve/main/da
>>> files = dl_manager.iter_files(files)
```

tập dữ liệu lớp.DownloadConfigdatasets.DownloadConfighttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/download_config.py#L10[{"name": "cache_dir", "val": ": typing.Union[str, pathlib.Path, NoneType] = None"}, {"name": "force_download", "val": ": bool = False"}, {"name": "resume_download", "val": ": bool = False"}, {"name": "local_files_only", "val": ": bool = False"}, {"name": "proxies", "val": ": typing.Optional[dict] = None"}, {"name": "user_agent", "val": ": typing.Optional[str] = None"}, {"name": "extract_compressed_file", "val": ": bool = False"}, {"name": "force_extract", "val": ": bool = False"}, {"name": "delete_extracted", "val": ": bool = False"}, {"name": "extract_on_the_fly", "val": ": bool = False"}, {"name": "use_etag", "val": ": bool = True"}, {"name": "num_proc", "val": ": typing.Optional[int] = None"}, {"name": "max_retries", "val": ": int = 1"}, {"name": "token", "val": ": typing.Union[str, bool, NoneType] = None"}, {"name": "storage_options", "val": ": dict = "}, {"name": "download_desc", "val": ": typing.Optional[str] = None"}, {"name": "disable_tqdm", "val": ": bool = False"}]- cache_dir (str or Path , optional) --
Specify a cache directory to save the file to (overwrite the
default cache dir).

- force_download (bool , defaults to False) --

Nếu Đúng, hãy tải xuống lại tệp ngay cả khi nó đã được lưu vào bộ nhớ đệm
thư mục bộ đệm.

- resume_download (bool , defaults to False) --
  Nếu Đúng , hãy tiếp tục tải xuống nếu tệp nhận được chưa đầy đủ
  thành lập.
- proxies (dict , optional) --
- user_agent (str , optional) --
  Chuỗi hoặc lệnh tùy chọn sẽ được thêm vào tác nhân người dùng trên điều khiển từ xa
  yêu cầu.
- extract_compressed_file (bool , defaults to False) --
  Nếu Đúng và đường dẫn trỏ đến tệp zip hoặc tar,
  giải nén tệp nén trong một thư mục dọc theo kho lưu trữ.
- force_extract (bool , defaults to False) --
  Nếu đúng khi extract_compression_file là True và kho lưu trữ
  đã được giải nén, hãy giải nén lại kho lưu trữ và ghi đè thư mục chứa nó
  được chiết xuất.
- delete_extracted (bool , defaults to False) --
  Whether to delete (or keep) the extracted files.
- extract_on_the_fly (bool , defaults to False) --
  Nếu Đúng, hãy giải nén các tệp nén trong khi chúng đang được đọc.
- use_etag (bool , defaults to True) --
  Có sử dụng tiêu đề phản hồi HTTP ETag để xác thực các tệp được lưu trong bộ nhớ đệm hay không.
- num_proc (int , optional) --
  Số lượng tiến trình khởi chạy để tải xuống các tệp song song.
- max_retries (int , default to 1) --
  Số lần thử lại yêu cầu HTTP nếu không thành công.
- token (str or bool , optional) --
  Chuỗi hoặc boolean tùy chọn để sử dụng làm mã thông báo Bearer
  cho các tập tin từ xa trên Datasets Hub. Nếu Đúng hoặc không được chỉ định, sẽ nhận được mã thông b
  ~/.huggingface .
- storage_options (dict , optional) --
  Các cặp khóa/giá trị sẽ được chuyển đến phần phụ trợ hệ thống tệp tập dữ liệu, nếu có.
- download_desc (str , optional) --
  Mô tả sẽ được hiển thị cùng với thanh tiến trình trong khi tải xuống tệp.
- disable_tqdm (bool , defaults to False) --

Có tắt thanh tiến trình tải xuống từng tệp riêng lẻ hay không
Cấu hình cho trình quản lý đường dẫn được lưu trong bộ nhớ cache của chúng tôi.

tập dữ liệu lớp.DownloadModedatasets.DownloadModehttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/download/download_manager.py#L50[{"name": "value", "val": ""}, {"name": "names", "val": " = None"}, {"name": "module", "val": " = None"}, {"name": "qualname", "val": " = None"}, {"name": "type", "val": " = None"}, {"name": "start", "val": " = 1"}]
Enum để biết cách xử lý dữ liệu và nội dung tải xuống có sẵn.

Chế độ mặc định là REUSE_DATASET_IF_EXISTS, sẽ sử dụng lại cả hai
tải xuống thô và tập dữ liệu đã chuẩn bị nếu chúng tồn tại.

Các chế độ thế hệ:

	Tải xuống	Tập dữ liệu
REUSE_DATASET_IF_EXISTS (default)	Reuse	Reuse
REUSE_CACHE_IF_EXISTS	Tái sử dụng	Tươi
FORCE_REDDOWNLOAD	Tươi	Tươi

Bộ dữ liệu xác minh.VerificationMode

bộ dữ liệu lớp.VerificationModedatasets.VerificationModehttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/utils/info_utils.py#L22[{"name": "value", "val": ""}, {"name": "names", "val": " = None"}, {"name": "module", "val": " = None"}, {"name": "qualname", "val": " = None"}, {"name": "type", "val": " = None"}, {"name": "start", "val": " = 1"}]
Enum chỉ định những bước kiểm tra xác minh nào sẽ chạy.

Chế độ mặc định là BASIC_CHECKS, chế độ này sẽ chỉ thực hiện các kiểm tra thô sơ để tránh sự chậm lại
khi tạo/tải xuống tập dữ liệu lần đầu tiên.

Các chế độ xác minh:

	Kiểm tra xác minh
ALL_CHECKS	Kiểm tra phân tách, tính duy nhất của các khóa mang lại trong trường hợp Máy Phát ĐiệnXây Dựng and the validity (number of files, checksums, etc.) of downloaded tập tin
BASIC_CHECKS (default)	Tương tự như ALL_CHECKS nhưng không kiểm tra các tệp đã tải xuống
KHÔNG_CHECKS	Không có

Splitsdatasets.SplitGenerator

tập dữ liệu lớp.SplitGeneratordatasets.SplitGeneratorhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/splits.py#L602[{"name": "name", "val": ": str"}, {"name": "gen_kwargs", "val": ": dict = "}]- name (str) --
Tên của phần phân chia mà trình tạo sẽ
tạo ra các ví dụ

- **gen_kwargs (additional keyword arguments) --
  Đối số từ khóa để chuyển tiếp tới phương thức DatasetBuilder._generate_examples của người xây dựng.0
  Xác định thông tin phân chia cho trình tạo.

Giá trị này nên được sử dụng làm giá trị trả về của
GeneratorBasedBuilder._split_generators .
Xem GeneratorBasedBuilder._split_generators để biết thêm thông tin và ví dụ
của việc sử dụng.

Ví dụ:

```
>>> datasets.SplitGenerator(
...name=datasets.Split.TRAIN,
...gen_kwargs={"split_key": "train", "files": dl_manager.download_and_extract(url)},
... )
```

bộ dữ liệu lớp.Splitdatasets.Splithttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/splits.py#L406[{"name": "name", "val": ""}]
Enum để phân chia tập dữ liệu.

Các bộ dữ liệu thường được chia thành các tập con khác nhau để sử dụng ở nhiều giai đoạn đào tạo và đánh giá.

- TRAIN : dữ liệu huấn luyện.
- VALIDATION : dữ liệu xác thực. Nếu có, điều này thường được sử dụng như evaluation data while iterating on a model (e.g. changing hyperparameters, model architecture, etc.).
- TEST : dữ liệu thử nghiệm. Đây là dữ liệu để báo cáo số liệu. Tiêu biểu bạn không muốn sử dụng điều này trong quá trình lặp lại mô hình vì bạn có thể quá phù hợp với nó.
- ALL : sự kết hợp của tất cả các phần tách tập dữ liệu được xác định.

Tất cả các phần tách, bao gồm cả các tác phẩm đều kế thừa từ bộ dữ liệu.SplitBase.

Xem hướng dẫn về cách chia tách để biết thêm thông tin.

Ví dụ:

```
>>> datasets.SplitGenerator(
...name=datasets.Split.TRAIN,
...gen_kwargs={"split_key": "train", "files": dl_manager.download_and extract(url)},
... ),
... datasets.SplitGenerator(
...name=datasets.Split.VALIDATION,
...gen_kwargs={"split_key": "validation", "files": dl_manager.download_and extract(url)},
... ),
... datasets.SplitGenerator(
...name=datasets.Split.TEST,
...gen_kwargs={"split_key": "test", "files": dl_manager.download_and extract(url)},
... )
```

bộ dữ liệu lớp.NamedSplitdatasets.NamedSplithttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/splits.py#L314[{"name": "name", "val": ""}]
Descriptor corresponding to a named split (train, test, ...).

Ví dụ:

Mỗi bộ mô tả có thể được kết hợp với bộ mô tả khác bằng cách sử dụng phép cộng hoặc lát cắt:

```
split = datasets.Split.TRAIN.subsplit(datasets.percent[0:25]) + datasets.Split.TEST
```

Phần chia kết quả sẽ tương ứng với 25% phần chia đoàn tàu được hợp nhất với
100% phần chia bài kiểm tra.

Một phần tách không thể được thêm hai lần, vì vậy những điều sau đây sẽ không thành công:

```
split = (
        datasets.Split.TRAIN.subsplit(datasets.percent[:25]) +
        datasets.Split.TRAIN.subsplit(datasets.percent[75:])
)# Error
split = datasets.Split.TEST + datasets.Split.ALL# Error
```

Các lát cắt chỉ có thể được áp dụng một lần. Vì vậy, những điều sau đây là hợp lệ:

```
split = (
        datasets.Split.TRAIN.subsplit(datasets.percent[:25]) +
        datasets.Split.TEST.subsplit(datasets.percent[:50])
)
split = (datasets.Split.TRAIN + datasets.Split.TEST).subsplit(datasets.percent[:50])
```

Nhưng điều này không hợp lệ:

```
train = datasets.Split.TRAIN
test = datasets.Split.TEST
split = train.subsplit(datasets.percent[:25]).subsplit(datasets.percent[:25])
split = (train.subsplit(datasets.percent[:25]) + test).subsplit(datasets.percent[:50])
```

bộ dữ liệu lớp.NamedSplitAlldatasets.NamedSplitAllhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/splits.py#L391[]
Phân chia tương ứng với sự kết hợp của tất cả các phân chia tập dữ liệu được xác định.

bộ dữ liệu lớp.ReadInstructiondatasets.ReadInstructionhttps://github.com/huggingface/

datasets/blob/4.2.0/src/datasets/arrow_reader.py#L456[{"name": "split_name", "val": ""}, {"name": "rounding", "val": " = None"}, {"name": "from_", "val": " = None"}, {"name": "to", "val": " = None"}, {"name": "unit", "val": " = None"}]
Hướng dẫn đọc một tập dữ liệu.

Ví dụ:

```
# Các dòng sau là tương đương:
ds = datasets.load_dataset('mnist', split='test[:33%]')
ds = datasets.load_dataset('mnist', split=datasets.ReadInstruction.from_spec('test[:33%]'))
ds = datasets.load_dataset('mnist', split=datasets.ReadInstruction('test', to=33, unit='%'))
ds = datasets.load_dataset('mnist', split=datasets.ReadInstruction(
'test', from_=0, to=33, unit='%'))

# Các dòng sau là tương đương:
ds = datasets.load_dataset('mnist', split='test[:33%]+train[1:-1]')
ds = datasets.load_dataset('mnist', split=datasets.ReadInstruction.from_spec(
'test[:33%]+train[1:-1]'))
ds = datasets.load_dataset('mnist', split=(
datasets.ReadInstruction('test', to=33, unit='%') +
datasets.ReadInstruction('train', from_=1, to=-1, unit='abs')))

# Các dòng sau là tương đương:
ds = datasets.load_dataset('mnist', split='test[:33%](pct1_dropremainder)')
ds = datasets.load_dataset('mnist', split=datasets.ReadInstruction.from_spec(
'test[:33%](pct1_dropremainder)'))
ds = datasets.load_dataset('mnist', split=datasets.ReadInstruction(
'test', from_=0, to=33, unit='%', rounding="pct1_dropremainder"))

# Xác thực 10 lần:
tests = datasets.load_dataset(
'mnist',
[datasets.ReadInstruction('train', from_=k, to=k+10, unit='%')
for k in range(0, 100, 10)])
trains = datasets.load_dataset(
'mnist',
[datasets.ReadInstruction('train', to=k, unit='%') + datasets.ReadInstruction('train', from_=k+10,
for k in range(0, 100, 10)])
```

from_specdatasets.ReadInstruction.from_spechttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/arrow_reader.py#L536[{"name": "spec", "val": ""}]- spec (str) --
Split(s) + optional slice(s) to read + optional rounding
nếu phần trăm được sử dụng làm đơn vị cắt. Một lát cắt có thể được chỉ định,
using absolute numbers (int) or percentages (int).0ReadInstruction instance.
Tạo một phiên bản ReadInstruction từ một thông số chuỗi.

Ví dụ:

kiểm tra: chia thử nghiệm.
kiểm tra + xác nhận: phân tách kiểm tra + phân tách xác thực.
test[10:]: test split, minus its first 10 records.
test[:10%]: first 10% records of test split.
test[:20%](pct1_dropremainder): first 10% records, rounded with the pct1_dropremainder rounding.
test[:-5%]+train[40%:60%]: first 95% of test + middle 20% of train.

to_absolutedatasets.ReadInstruction.to_absolutehttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/arrow_reader.py#L608[{"name": "name2len", "val": ""}]- name2len (dict) --
Liên kết tên phân chia với số ví dụ.0danh sách các phiên bản _AbsoluteInstruction
(corresponds to the + in spec).
Dịch hướng dẫn thành một danh sách các hướng dẫn tuyệt đối.

Những chỉ dẫn tuyệt đối đó sau đó sẽ được cộng lại với nhau.

Versiondatasets.Version

bộ dữ liệu lớp.Versiondatasets.Versionhttps://github.com/huggingface/datasets/blob/4.2.0/src/datasets/utils/version.py#L30[{"name": "version_str", "val": ": str"}, {"name": "description", "val": ": typing.Optional[str] = None"}, {"name": "major", "val": ": typing.Union[str, int, NoneType] = None"}, {"name": "minor", "val": ": typing.Union[str, int, NoneType] = None"}, {"name": "patch", "val": ": typing.Union[str, int, NoneType] = None"}]- version_str (str) --
Phiên bản tập dữ liệu.

- description (str) --
  Mô tả về những gì mới trong phiên bản này.
- major (str) --

- minor (str) --
- patch (str) --0

  Phiên bản tập dữ liệu MAJOR.MINOR.PATCH .

Ví dụ:

```
>>> VERSION = datasets.Version("1.0.0")
```