Tải tập dữ liệu từ Hub

Việc tìm kiếm các bộ dữ liệu chất lượng cao có thể tái tạo và truy cập được có thể khó khăn. Một trong
Mục tiêu chính của bộ dữ liệu là cung cấp một cách đơn giản để tải tập dữ liệu ở bất kỳ định dạng hoặc loại n
cách dễ nhất để bắt đầu là khám phá tập dữ liệu hiện có trên Hugging Face Hub - một
bộ sưu tập dữ liệu hướng đến cộng đồng cho các nhiệm vụ trong NLP, thị giác máy tính và âm thanh - và sử
Bộ dữ liệu để tải xuống và tạo tập dữ liệu.

Hướng dẫn này sử dụng bộ dữ liệu rotten_tomatoes và MInDS-14, nhưng vui lòng tải bất kỳ bộ dữ liệu nào
tập dữ liệu bạn muốn và làm theo. Hãy truy cập Hub ngay bây giờ và tìm tập dữ liệu cho nhiệm vụ của bạn!

Tải tập dữ liệu

Trước khi bạn dành thời gian tải xuống một tập dữ liệu, việc nhanh chóng có được một số thông tin chung th
thông tin về một tập dữ liệu. Thông tin của tập dữ liệu được lưu trữ bên trong DatasetInfo và có thể
bao gồm các thông tin như mô tả tập dữ liệu, tính năng và kích thước tập dữ liệu.

Use the load_dataset_builder() function to load a dataset builder and inspect a dataset's
thuộc tính mà không cam kết tải xuống:

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

# Kiểm tra mô tả tập dữ liệu
>>> ds_builder.info.description
Bộ dữ liệu đánh giá phim. Đây là tập dữ liệu chứa 5.331 dương tính và 5.331 âm tính được xử lý

# Kiểm tra các tính năng của tập dữ liệu
>>> ds_builder.info.features
{'label': ClassLabel(names=['neg', 'pos']),
 'text': Value('string')}
```

If you're happy with the dataset, then load it with load_dataset():

```
>>> from datasets import load_dataset

>>> dataset = load_dataset("cornell-movie-review-data/rotten_tomatoes", split="train")
```

Chia tách

Phần tách là một tập hợp con cụ thể của tập dữ liệu như train và test . Liệt kê tên phân chia của tập dữ liệu v
the get_dataset_split_names() function:

```
>>> from datasets import get_dataset_split_names

>>> get_dataset_split_names("cornell-movie-review-data/rotten_tomatoes")
['train', 'validation', 'test']
```

Sau đó, bạn có thể tải một phần phân chia cụ thể bằng tham số phân tách. Đang tải phần chia dữ liệu trả về
một đối tượng Tập dữ liệu:

```
>>> from datasets import load_dataset

>>> dataset = load_dataset("cornell-movie-review-data/rotten_tomatoes", split="train")
>>> dataset
Dataset({
    features: ['text', 'label'],
    num_rows: 8530
})
```

Nếu bạn không chỉ định phần tách,  Bộ dữ liệu sẽ trả về một đối tượng DatasetDict:

```
>>> from datasets import load_dataset

>>> dataset = load_dataset("cornell-movie-review-data/rotten_tomatoes")
DatasetDict({
    train: Dataset({
        features: ['text', 'label'],
        num_rows: 8530
    })
    validation: Dataset({
        features: ['text', 'label'],
        num_rows: 1066
    })
    test: Dataset({
        features: ['text', 'label'],
        num_rows: 1066
    })
})
```

Cấu hình

Một số bộ dữ liệu chứa một số bộ dữ liệu phụ. Ví dụ: bộ dữ liệu MInDS-14 có một số
tập dữ liệu con, mỗi tập dữ liệu chứa dữ liệu âm thanh bằng một ngôn ngữ khác nhau. Các tập dữ liệu con n
được gọi là cấu hình hoặc tập hợp con và bạn phải chọn rõ ràng một cấu hình khi tải
tập dữ liệu. Nếu bạn không cung cấp tên cấu hình,  Bộ dữ liệu sẽ đưa ra ValueError và
nhắc nhở bạn chọn cấu hình.

Use the get_dataset_config_names() function to retrieve a list of all the possible configurations
có sẵn cho tập dữ liệu của bạn:

```
>>> from datasets import get_dataset_config_names

>>> configs = get_dataset_config_names("PolyAI/minds14")
>>> print(configs)
['cs-CZ', 'de-DE', 'en-AU', 'en-GB', 'en-US', 'es-ES', 'fr-FR', 'it-IT', 'ko-KR', 'nl-NL', 'pl-PL'
```

Sau đó tải cấu hình bạn muốn:

```
>>> from datasets import load_dataset

>>> mindsFR = load_dataset("PolyAI/minds14", "fr-FR", split="train")
```