Tạo tập dữ liệu âm thanh

Bạn có thể chia sẻ tập dữ liệu với nhóm của mình hoặc với bất kỳ ai trong cộng đồng bằng cách tạo tập dữ liệ
kho lưu trữ trên Hugging Face Hub:

```
từ tập dữ liệu nhập Load_dataset

dataset = load_dataset("<username>/my_dataset")
```

Có một số phương pháp để tạo và chia sẻ tập dữ liệu âm thanh:

- Create an audio dataset from local files in python with Dataset.push_to_hub(). This is an
  cách dễ dàng chỉ cần một vài bước trong python.
- Tạo kho lưu trữ tập dữ liệu âm thanh bằng trình tạo AudioFolder. Đây là mã không có mã
  giải pháp để nhanh chóng tạo tập dữ liệu âm thanh với hàng nghìn tệp âm thanh.

[!TIP]
Bạn có thể kiểm soát quyền truy cập vào tập dữ liệu của mình bằng cách yêu cầu người dùng chia sẻ liên l
thông tin đầu tiên. Hãy xem hướng dẫn về bộ dữ liệu Gated để biết thêm thông tin về cách
kích hoạt tính năng này trên Hub.

Tệp cục bộ

You can load your own dataset using the paths to your audio files. Use the cast_column()
để lấy một cột đường dẫn tệp âm thanh và chuyển nó sang tính năng Âm thanh:

```
>>> audio_dataset = Dataset.from_dict({"audio": ["path/to/audio_1", "path/to/audio_2", ..., "path/
>>> audio_dataset[0]["audio"]
<datasets.features._torchcodec.AudioDecoder object at 0x11642b6a0>
```

Then upload the dataset to the Hugging Face Hub using Dataset.push_to_hub():

```
audio_dataset.push_to_hub("<username>/my_dataset")
```

Điều này sẽ tạo một kho lưu trữ tập dữ liệu chứa tập dữ liệu âm thanh của bạn:

```
my_dataset/
 README.md
 dữ liệu/
        train-00000-of-00001.parquet
```

Thư mục âm thanh

AudioFolder là trình tạo tập dữ liệu được thiết kế để tải nhanh tập dữ liệu âm thanh với một số nghìn tệp âm thanh mà không yêu cầu bạn phải viết bất kỳ mã nào.

[!TIP]
Hãy xem hệ thống phân cấp Mẫu phân chia để tìm hiểu thêm về cách AudioFolder tạo phân chia tập dữ liệu dựa trên cấu trúc kho lưu trữ tập dữ liệu của bạn.

AudioFolder tự động suy ra nhãn lớp của tập dữ liệu của bạn dựa trên tên thư mục.
Lưu trữ tập dữ liệu của bạn trong cấu trúc thư mục như:

```
folder/train/dog/golden_retriever.mp3
folder/train/dog/german_shepherd.mp3
folder/train/dog/chihuahua.mp3

folder/train/cat/maine_coon.mp3
thư mục/train/cat/bengal.mp3
folder/train/cat/birman.mp3
```

Nếu tập dữ liệu tuân theo cấu trúc AudioFolder thì bạn có thể tải nó trực tiếp bằng load_dataset():

```
>>> from datasets import load_dataset

>>> dataset = load_dataset("username/dataset_name")
```

This is equivalent to passing audiofolder manually in load_dataset() and the directory in dữ liệu_dir :

```
>>> dataset = load_dataset("audiofolder", data_dir="/path/to/folder")
```

Bạn cũng có thể sử dụng thư mục âm thanh để tải tập dữ liệu liên quan đến nhiều phần tách. Để làm như vậ
thư mục nên có cấu trúc như sau:

```
folder/train/dog/golden_retriever.mp3
folder/train/cat/maine_coon.mp3
folder/test/dog/german_shepherd.mp3
thư mục/test/cat/bengal.mp3
```

> [!WARNING]
> Nếu tất cả các tập tin âm thanh được chứa trong một thư mục duy nhất hoặc nếu chúng không cùng cấp
> cấu trúc thư mục, cột nhãn sẽ không được thêm tự động. Nếu bạn cần thì hãy đặt
> drop_labels=False explicitly.

Nếu có thông tin bổ sung mà bạn muốn đưa vào về tập dữ liệu của mình, như chú thích văn bản hoặc
các hộp giới hạn, hãy thêm nó dưới dạng tệp siêu dữ liệu.csv trong thư mục của bạn. Điều này cho phép bạn
bộ dữ liệu cho các tác vụ thị giác máy tính khác nhau như chú thích văn bản hoặc phát hiện đối tượng. bạn c
cũng sử dụng tệp JSONL siêu dữ liệu.jsonl hoặc tệp Parquet siêu dữ liệu.parquet .

```
thư mục/train/metadata.csv
thư mục/tàu/0001.mp3
thư mục/tàu/0002.mp3
thư mục/tàu/0003.mp3
```

Bạn cũng có thể nén các tệp âm thanh của mình và trong trường hợp này, mỗi tệp zip phải chứa cả hai tệp â
và siêu dữ liệu

```
thư mục/train.zip
thư mục/test.zip
thư mục/xác thực.zip
```

Tệp siêu dữ liệu.csv của bạn phải có trường file_name hoặc *_file_name liên kết các tệp âm thanh
với siêu dữ liệu của họ:

```
tên_tệp,tính năng bổ sung
0001.mp3,Đây là giá trị đầu tiên của tính năng văn bản bạn đã thêm vào tệp âm thanh của mình
0002.mp3,Đây là giá trị thứ hai của tính năng văn bản bạn đã thêm vào tệp âm thanh của mình
0003.mp3,Đây là giá trị thứ ba của tính năng văn bản bạn đã thêm vào tệp âm thanh của mình
```

hoặc sử dụng siêu dữ liệu.jsonl:

```
{"file_name": "0001.mp3", "additional_feature": "This is a first value of a text feature you added
{"file_name": "0002.mp3", "additional_feature": "This is a second value of a text feature you adde
{"file_name": "0003.mp3", "additional_feature": "This is a third value of a text feature you added
```

Ở đây file_name phải là tên của tệp âm thanh bên cạnh tệp siêu dữ liệu. Hơn
nói chung, nó phải là đường dẫn tương đối từ thư mục chứa siêu dữ liệu đến âm thanh
tài liệu.

Có thể trỏ tới nhiều âm thanh trong mỗi hàng trong tập dữ liệu của bạn, chẳng hạn nếu cả hai
đầu vào và đầu ra của bạn là các tệp âm thanh:

```
{"input_file_name": "0001.mp3", "output_file_name": "0001_output.mp3"}
{"input_file_name": "0002.mp3", "output_file_name": "0002_output.mp3"}
{"input_file_name": "0003.mp3", "output_file_name": "0003_output.mp3"}
```

Bạn cũng có thể xác định danh sách các tập tin âm thanh. Trong trường hợp đó bạn cần đặt tên cho trường f
*_file_names . Đây là một ví dụ:

```
{"recordings_file_names": ["0001_r0.mp3", "0001_r1.mp3"], label: "same_person"}
{"recordings_file_names": ["0002_r0.mp3", "0002_r1.mp3"], label: "same_person"}
{"recordings_file_names": ["0003_r0.mp3", "0003_r1.mp3"], label: "different_person"}
```

Bộ dữ liệu Web

Định dạng WebDataset dựa trên kho lưu trữ TAR và phù hợp với các bộ dữ liệu âm thanh lớn.
Indeed you can group your audio files in TAR archives (e.g. 1GB of audio files per TAR
archive) and have thousands of TAR archives:

```
thư mục/train/00000.tar
thư mục/train/00001.tar
thư mục/train/00002.tar
None
```

Trong kho lưu trữ, mỗi ví dụ được tạo từ các tệp có chung tiền tố:

```
e39871fd9fd74f55.mp3
e39871fd9fd74f55.json
f18b91585c4d3f3e.mp3
f18b91585c4d3f3e.json
ede6e66b2fb59aab.mp3
ede6e66b2fb59aab.json
ed600d57fcee4f94.mp3
ed600d57fcee4f94.json
None
```

Bạn có thể đặt nhãn/chú thích/hộp giới hạn cho tệp âm thanh của mình bằng cách sử dụng tệp JSON hoặc và
ví dụ.

Load your WebDataset and it will create on column per file suffix (here "mp3" and "json"):

```
>>> from datasets import load_dataset

>>> dataset = load_dataset("webdataset", data_dir="/path/to/folder", split="train")
>>> dataset[0]["json"]
{"transcript": "Hello there !", "speaker": "Obi-Wan Kenobi"}
```

Cũng có thể có một số tệp âm thanh cho mỗi ví dụ như thế này:

```
e39871fd9fd74f55.input.mp3
e39871fd9fd74f55.output.mp3
e39871fd9fd74f55.json
f18b91585c4d3f3e.input.mp3
f18b91585c4d3f3e.output.mp3
f18b91585c4d3f3e.json
None
```

Để biết thêm chi tiết về định dạng WebDataset và thư viện python, vui lòng kiểm tra Tài liệu WebDataset.