Datasets: A Community Library for NLP

What it is | Why it is needed

Paper: Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021)

Motivation

Dataset paradigms for NLP

Hansard (1988)	Statistical translation
PTB (1993)	Syntactic modelling
UD (2016)	Cross-lingual annos

Modern empirical NLP uses blend of datasets

C4 (2020)	SQuAD (2016)	GLUE (2019)
pre-training	fine-tuning	evaluation

Dataset diversity brings challenges

How can we provide practitioners with:

- a consistent interface, regardless of scale
- clear versioning information
- information about construction (e.g. Datasheets)

Datasets aims to meet these challenges. Goals:

Usability & standardisation | Efficiency | Community & docs

The library is released under Apache 2.0 license

References:
(Hansard) P. F. Brown et al. "A statistical approach to language translation", COLING (1988)
(PTB) M. P. Marcus et al. "Building a large annotated corpus of English: The Penn Treebank", Computational Linguistics (1993)
(UD) J. Nivre et al., "Universal dependencies v1: A multilingual treebank collection", LREC (2016)
(C4) C. Raffel et al., "Exploring the limits of transfer learning with a unified text-to-text transformer", JMLR (2020)
(SQuAD) P. Rajpurkar et al., "SQuAD: 100, 000+ Questions for Machine Comprehension of Text", EMNLP (2016)
(GLUE) A. Wang et al., "GLUE: A multi-task benchmark and analysis platform for natural language understanding", ICLR (2019)
(Datasheets) T. Gebru et al., "Datasheets for datasets", Communications of the ACM (2021)

Library Design

A little tour

Datasets are loaded with a global identity

```
dataset = load_dataset("boolq")
```

Datasets have metadata/features schema

```
dataset.info, dataset.features
```

Datasets enable lazy loads via slicing

```
dataset["train"][start:end]
```

Parallel/batch processing of data points

```
# Apply "tokenise" function
tokenized = dataset.map(tokenise,
                        num_proc=32)
```

1. Dataset retrieval & building

No hosting of raw data
(access from original authors)
Builder module for each dataset
The builder converts raw data
into a common representation

2. Data point representation

Tables with typed columns
Standard/NLP types supported:
int, float, string, blob, dict, list
named categorical labels
multi-dimensional arrays

3. In-memory access

Use Apache Arrow for access
Memory-mapping for big data
Zero-copy reads to ML libraries
(PyTorch, TensforFlow,...)

4. User Processing

Low processing at download
Support shuffling, splitting etc.
Map allows arbitrary functions
for creating in-memory tables
Enables batched/parallel tasks

Flow

Dataset request → Download from host → Builder → Access to mem-mapped table → Vectorised processing

References:
Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021)
C. Clark et al. "BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions", NAACL-HLT (2019)
(Apache Arrow) https://arrow.apache.org/

Dataset documentation and search

The Dataset Hub enables dataset exploration:

Structured tags	tasks, licenses, languages etc.
Data Card	technical & broader context
List of models	(those trained on the dataset)

Choosing a dataset

Structured tags enable faceted search

Filter (e.g. task: extractive-qa, license: MIT)

Using a dataset

The Data Card provides details:

splits | size on disk | descriptions of sample fields

Data Card as living document

The Data Card is community maintained

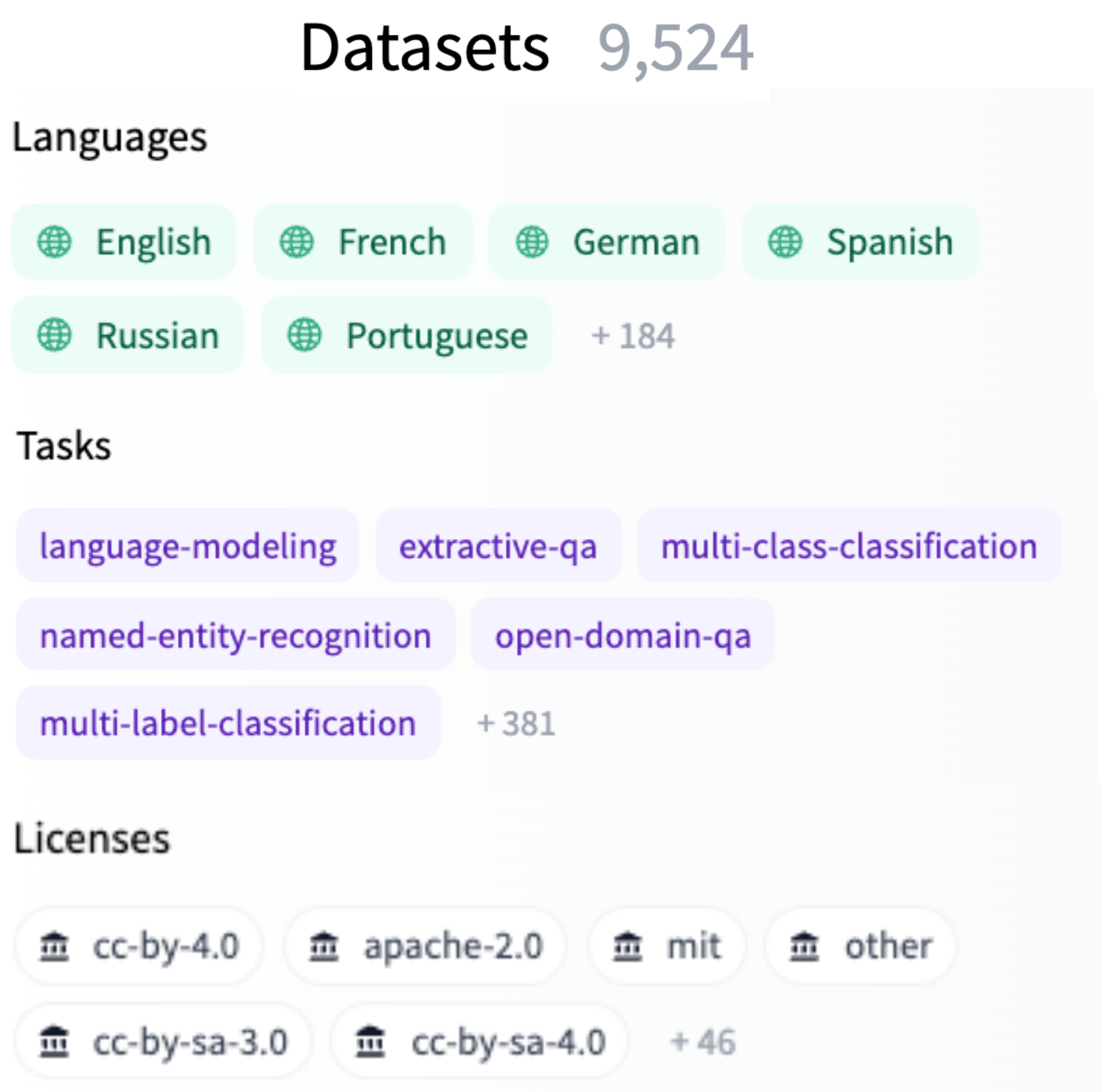
Practitioners can flag issues

annotation artifacts | issues with splits | biases

References:
Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021)
(Dataset Hub) https://huggingface.co/datasets/
(Data Card) A. McMillan-Major et al., "Reusable Templates and Guides For Documenting Datasets and Models for Natural Language Processing and Generation: A Case Study of the HuggingFace and GEM Data and Model Cards", GEM (2021)

Dataset usage and use-cases

https://hf.co/datasets (Sep. 2022)



Case Study: *N*-task Benchmarks

GLUE (+ others) popularised *N*-task benchmarks

Eleuther AI | lm-evaluation-harness (200+ tasks)

Case Study: Reproducible Shared Tasks

NLP - history of long-lived benchmarks (CoNLL)

GEM workshop (many tasks, 1 line of code)

Case Study: Robustness Evaluation

Robustness remains a key issue for NLP

Robustness Gym uses the library data interface

References
Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021)
(GLUE) A. Wang et al., "GLUE: A multi-task benchmark and analysis platform for natural language understanding", ICLR (2019)
https://github.com/EleutherAI/lm-evaluation-harness
E. F. Sang et al., "Introduction to the CoNLL-2000 shared task: Chunking", arxiv (2000)
S. Gehrmann et al., "The gem benchmark: Natural language generation, its evaluation and metrics", arxiv (2021)
K. Goel et al., "Robustness gym: Unifying the nlp evaluation landscape", arxiv (2021)

Further functionality

Streaming

Some datasets cannot fit on disk

Streaming mode: buffer data on the fly

Support for the map() primitive

Indexing

Support for search indexes:

FAISS

ElasticSearch

Metrics

Interface to match metrics and datasets

But, shortly replaced by Evaluate

Data Preview

Interactive data viewer

Dataset Preview

Go to dataset viewer

Subset: cola

Split: train

sentence (string)	label (class label)	idx (int32)
"The critics laughed the play off the stage."	1 (acceptable)	10
"The pond froze solid."	1 (acceptable)	11
"Bill rolled out of the room."	1 (acceptable)	12
"The gardener watered the flowers flat."	1 (acceptable)	13
"The gardener watered the flowers."	1 (acceptable)	14
"Bill broke the bathtub into pieces."	1 (acceptable)	15
"Bill broke the bathtub."	1 (acceptable)	16
"They drank the pub dry."	1 (acceptable)	17
"They drank the pub."	0 (unacceptable)	18
"The professor talked us into a stupor."	1 (acceptable)	19
"The professor talked us."	0 (unacceptable)	20

References/image credits:
Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021)
(FAISS) J. Johnson et al., "Billion-scale similarity search with gpus", IEEE Transactions on Big Data (2019)
(ElasticSearch) https://github.com/elastic/elasticsearch
(Data Preview) Screenshot from https://huggingface.co/datasets/glue, Sep 2022

Prior work

Linguistic Data Consortium (1992)

Distributes and manages language data

OntoNotes (2006)

Annotate multiple tasks over one corpus

Universal Dependencies (2016)

Cross-lingual treebank annotations

🤗 Datasets aims for content-agnostic access to a range of datasets

NLTK (2006)

Simplified downloading core datasets

SpaCy (2017)

Simple download interface

Deep learning libraries

TorchText

TensorFlow-Datasets

🤗 Datasets began as a fork

🤗 Datasets aims to:

- offer framework independence
- provide general-purpose tabular API
- prioritise community management
- cover long-tail of tasks/languages

References:
Q. Lhoest et al., "Datasets: A community library for natural language processing", EMNLP Demo (2021)
(Linguistic Data Consortium) https://www.ldc.upenn.edu/about
E. Hovy et al., "OntoNotes: the 90% solution", HLT-NAACL (2006)
(UD) J. Nivre et al., "Universal dependencies v1: A multilingual treebank collection", LREC (2016)
(NLTK) S. Bird, "NLTK: the natural language toolkit", COLING/ACL Interactive Presentation Sessions (2006)
(SpaCy) M. Honnibal et al., "spaCy 2: Natural Language Understanding with Bloom Embeddings, Convolutional Neural Networks and Incremental Parsing", GitHub (2017)
(TorchText) https://pytorch.org/text/stable/index.html
(TensorFlow-Datasets) https://www.tensorflow.org/datasets