

# Extracted Features Dataset

## Page-level features from 15.7 million volumes

A great deal of useful research can be performed non-consumptively with pre-extracted features. For this reason, we've prepared a data export of features for 15.7 million volumes of the HathiTrust Digital Library.

*Features* are notable or informative characteristics of the text. We have processed a number of useful features, including part-of-speech tagged token counts, header and footer identification, and various line-level information. These are provided *per-page*. Providing token information at the page-level makes it possible to separate text from paratext; for instance, a researcher may use the information to identify publishers' ads at the back of a book. For cleaner text, headers and footers are also identified distinctly from page content. The specific features that we extract for each page are described in more detail below.

The most useful extracted feature that we provide is the token (unigram) count, on a per-page basis. Term counts are specific to the part-of-speech usage for that term, so that a term used as both a noun and a verb, for example, will have separate counts provided for both these modalities of its use. We also include line information, such as the number of lines with text on each page, and a count of characters that start and end lines on each page. This information can illuminate genre and volume structure: for instance, it helps distinguish poetry from prose, or body text from an index.

Review the documentation below or jump straight to the [downloads](#).

## Attribution

Boris Capitanu, Ted Underwood, Peter Organisciak, Timothy Cole, Maria Janina Sarol, J. Stephen Downie (2016). The HathiTrust Research Center Extracted Feature Dataset (1.0) [Dataset]. HathiTrust Research Center, <http://dx.doi.org/10.13012/J8X63JT3>.

This feature dataset is released under a [Creative Commons Attribution 4.0 International License](#).

*This is the full release (Nov 2016). The earlier, public-domain subset (Spring 2015) is [still available](#).*

## Data Stats

# of volumes	15,722,079
# of pages	5,787,519,444
# of tokens	2,449,739,213,773
# of IC volumes	9,914,509
# of IC pages	3,005,448,348
# of IC tokens	1,777,793,828,310
# of PD volumes	5,807,570
# pd pages	2,602,212,586
# pd tokens	1,197,838,539,662

## Feature File Documentation

The HTRC Extracted Features provides a small amount of metadata in addition to the quantitative features.

### Metadata

A significant amount of bibliographic metadata for identifying the volume is included in this dataset. See also: "[Where can I find detailed bibliographic metadata?](#)".

**volumeIdentifier:** A unique identifier for the current volume. This is the same identifier used in the HathiTrust and HathiTrust Research Center corpora.

**schemaVersion:** A version identifier for the format and structure of this metadata object. *metadata.schemaVersion* is separate from *features.schemaVersion* below.

**dateCreated:** The time this metadata object was processed. *metadata.dateCreated* is not necessarily the same as the *features.dateCreated* below.

**accessProfile:** The original source of a volume.

**title:** Title of the volume.

**pubDate:** The publication year.

**pubPlace:** The publication location.

**language:** The primary language of the volume.

**genre:** The genre of the volume.

**issuance:** The bibliographic level of a volume

### Features

#### Volume-Level Features

**schemaVersion:** A version identifier for the format and structure of the feature data (HTRC generated).

**dateCreated:** The time the batch of metadata was processed and recorded (HTRC generated).

**pageCount:** The number of pages in the volume.

**pages:** An array of JSON objects, each representing a page of the volume.

#### Page-Level Features

Pages are contained within volumes, they have a sequence number, and information about their header, body, and footer.

#### Page-level information

**seq:** The sequence number. [More details on this value](#).

**tokenCount:** The total number of tokens on the page.

**lineCount:** The total number of non-empty lines on the page.

**emptyLineCount:** The total number of empty lines on the page.

**typeOfResource:** The format type of a volume.

**names:** The personal and corporate names associated with a volume.

**imprint:** The place of publication, publisher, and publication date of the given volume.

**enumerationChronology:** Contains issue-specific information in unstructured form (number, publication date)

**governmentDocument:** Denotes if a volume is a government document.

**rightsAttributes:** The rights attributes for a volume.

**hathiTrustRecordNumber:** The unique record number for the volume in the HathiTrust Digital Library.

**htBibUrl:** The HathiTrust Bibliographic API call for the volume.

**handleUrl:** The persistent identifier for the given volume.

**sourceInstitution:** The original institution who contributed the volume.

**sourceInstitutionRecordNumber:** The unique record number for the volume from its original institution.

**oclc:** The array of [OCLC](#) number(s).

**isbn:** The International Standard Book Number for a volume.

**issn:** The International Standard Serial Number for a volume.

**lccn:** The Library of Congress Call Number for a volume.

**classification:** The call number supplied by the originating library.

**bibliographicFormat:** The format of a volume's bibliography.

**lastUpdatePage:** The date this page was last updated.

**sentenceCount:** Total number of sentences identified on the page using OpenNLP. [Details on parsing.](#)

**languages:** Automatically inferred language likelihood for the page, Shuyo Nakatani's [Language Detection](#) library. [Language code reference.](#)

## Header, Body, and Footer information

The fields for *header*, *body*, and *footer* are the same, but pertain to different parts of the page. [Read about the differences between the sections.](#)

**tokenCount:** The total number of tokens in this page section.

**lineCount:** The number of lines containing characters of any kind in this page section. This pertains to the layout of the page; for sentence counts, see the *sentenceCount* field.

**emptyLineCount:** The number of lines without text in this page section.

**sentenceCount:** The number of sentences found in the text in this page section, parsed using OpenNLP.

**tokenPosCount:** An unordered list of all tokens (characterized by part of speech using OpenNLP), and their corresponding frequency counts, in this page section. Tokens are case-sensitive, so a capitalized "Rose" is shown as a separate token. There will be separate counts, for instance, for "rose" (noun) and "rose" (verb). Words separated by a hyphen across a line break are rejoined. No other data cleaning or OCR correction was performed. [Details on POS parsing and types of tags used.](#)

**beginCharCounts:** Aggregated frequency counts of the first non-whitespace character on each line. (Renamed in v.1, previously *beginLineChars*).

**endCharCount:** Count of the last character on each line in this page section (ignoring whitespace). (Renamed in v.1, previously *endLineChars*).

**capAlphaSeq:** The longest length of the alphabetical sequence of capital characters starting a line. (Body only).

---

## Get the Data

The data is accessible using rsync. Rsync should be installed already on your Mac or Linux system; Windows users can use it through [Cygwin](#).

### Sample Files

A sample of 100 extracted feature files is available for download through your browser: [sample.zip](#).

Also, thematic collections are available to download: [DocSouth](#) (87 volumes), [EEBO](#) (355 volumes), [ECCO](#) (505 volumes).

### Rsync

Rsync will synchronize files from our servers to your system. The Rsync module for the extracted features dataset is at **`data.analytics.hathitrust.org::features/`**.

*Note: Make sure to include the final period (.) when running your command in order to sync the files to your current directory, or else provide the path to the local directory of your choosing where you would like the files to be synced to.*

It is possible to sync any feature file in the following manner:

```
rsync -av data.analytics.hathitrust.org::features/{PATH-TO-FILE} .
```

A full listing of all the files is available in the following location:

```
rsync -azv data.analytics.hathitrust.org::features/listing/htrc-ef-all-files.txt .
```

To sync paths listed in text file:

```
rsync -av --files-from FILE.TXT data.analytics.hathitrust.org::features/ .
```

The data is kept in a pairtree directory structure, allowing you to infer the location of any file based on its HathiTrust volume identifier. Rather than learning the pairtree specification, we recommend using our command line [htid2rsync](#) tool:

```
htid2rsync {VOLUMEID} | rsync --files-from - data.analytics.hathitrust.org::features/
```

You can also use the [workset download script generator](#) to create a shell script that lists the rsync commands for every volume in an HTRC workset, and then run it:

```
sh <name of file>.sh
```

You can sync into a single folder, throwing away the directory structure, by adding `--no-relative` to the rsync command:

```
rsync -av --no-relative --files-from FILE.TXT data.analytics.hathitrust.org::features/ .
```

## Tips

- Not including a path will sync **all** files: this is possible, but remember that the *data is 4 Terabytes!* Only download all of it if you know what you are doing.
- The fastest way to sync is by pointing at exactly the files that you want. Use the `htrc-ef-all-files.txt` file if you want the paths to all the EF files.
- Have a custom list of volumes and want to get the file paths for it? We have created two options to help:
  - Uploading the volume list to the HTRC Portal as a workset allows you to download a shell script. [Details](#).
  - The [HTRC Feature Reader](#) installs a command line utility, `htid2rsync`, and Python functions for doing the same. [Details](#).

## Workset Download

To download the Extracted Features data for a specific workset in HTRC Analytics, there is an algorithm that generates the Rsync download script, the Extracted Features Download Helper.

## Workset Builder 2.0

Extracted Features files can also be downloaded for search results in [Workset Builder 2.0](#). First, enter search terms for a desired set of volumes. Once results are returned, apply any filtering for the results to remove any volumes for which you do not want Extracted Features files. Once your result set includes all of the volumes you'd like Extracted Features files for, click "JSON Extracted Features (ZIP)" under the "Export Search Results" heading at the top of the left sidebar. Clicking this link will start a download of the Extracted Features files for all volumes in your results. Since this process is fetching and compressing files, and result sets can be large, it may take a few moments to start and finish your download.

Alternatively, if you'd like to download Extracted Features files for individual volumes, you can click the link "Download Extracted Features (complete volume)" under each volume in the search results. This will download uncompressed Extracted Features for the given volume, in JSON.

## HTRC Feature Reader

We provide a Python library called the HTRC Feature Reader which simplifies many of the activities that you may want to perform with EF Dataset.

The best way to start with the HTRC Feature Reader is to read the introductory [tutorial](#).

## Acknowledgements

This research was supported in part by Lilly Endowment, Inc., through its support for the Indiana University Pervasive Technology Institute, and in part by the Indiana METACyt Initiative. The Indiana METACyt Initiative at IU is also supported in part by Lilly Endowment, Inc.

This release has been made possible, in part, by the National Endowment for the Humanities: Celebrating 50 Years of Excellence. Any views, findings, conclusions, or recommendations expressed in this release do not necessarily represent those of the National Endowment for the Humanities.

## Questions

### How are tokens parsed?

Hyphenation of tokens at end of line was corrected using custom code. [Apache OpenNLP](#) was used for sentence segmentation, tokenization, and part of speech (POS) tagging. No additional data cleaning or OCR correction was performed.

OpenNLP uses [the Penn Treebank POS tags](#).

### Can I use the page sequence as a unique identifier?

The `seq` value is always sequential from the start. Each scanned page of a volume has a unique sequence number, but it is specific to the *current* version of the full text. In theory, updates to the OCR that add or remove pages will change the sequence. The practical likelihood of changes in the sequence is low, but uses of the page as an id should be cautious.

A future release of this data will include persistent page identifiers that remain unchanged even when page sequence changes.

## Where is the bibliographic metadata? Who wrote the book?; When was it published, etc.?

This dataset is foremost an extracted features dataset, with minimal metadata included as a convenience. For additional metadata information, i.e. subject classifications, etc., HT offers [Hathifiles](#), which can be paired to our feature dataset through the volume *id* field.

The metadata that *is* included in this data includes MARC metadata from HathiTrust and additional information from Hathifiles:

- imprint: extracted from [MARC field 260](#), with MARC 260 subfield \$a pulled from the HathiTrust MARC record, and 260 \$b and 260 \$c from Hathifiles.
- language: extracted from [MARC control field 008](#) from Hathifiles.
- pubDate: extracted from Hathifiles. See also: [details on HathiTrust's rights-determination](#).
- oclc: OCLC number extracted from Hathifiles.

Additionally, schemaVersion and dateCreated are specific to this feature dataset.

## What do I do with beginning- or end-of-line characters?

The characters at the start and end of a line can be used to differentiate text from [paratext](#) at a page level. For instance, index lines tend to begin with capitalized letters and end with numbers. Likewise, lines in a table of contents can be identified through arabic or roman numerals at the start of a line.

## What is the difference between the header, body, and footer sections?

Because repeated headers and footers can distort word counts in a document, but also help identify document parts, we attempt to identify repeated lines at the top or bottom of a page and provide separate token counts for those forms of paratext. The "header" and "footer" sections will also include tokens that are page numbers, catchwords, or other short lines at the very top or bottom of a page. Users can of course ignore these divisions by aggregating the token counts for header, body, and footer sections.

## Known Issues

The current EF dataset has two known issues:

- Chinese and Japanese tokens sometimes have a zero-width space character, `\u200b`, at the start or end of the token.
- The character count information is under the keys 'beginCharCounts' and 'endCharChount'. Past and future versions will use the correct beginLineChars and endLineChars.

*Has a fix been released yet? No. Watch this space or adapt your code.*

## Contact Us

[htrc-help@hathitrust.org](mailto:htrc-help@hathitrust.org)

## Tools

If you've built tools or scripts for processing our data, let us know and we'll feature them here!

## Projects

Let us know about your projects and we'll link to them here.