Extracted Features [v.2.0]

Page-level features from 17.1 million volumes

HTRC Extracted Features 2.0 is the most current version of a derived dataset consisting of metadata and data elements extracted from volumes in the HathiTrust Digital Library. The dataset is composed of 17+ million JSON files representing a snapshot of the HathiTrust corpus from February 2020.

This documentation describes the structure and data in the HTRC Extracted Features 2.0 files for users of those files. The specific features extracted are described in more detail below.

You can also refer to technical documentation of the Extracted Features 3.0 JSON-LD schema here. The schema was developed collaboratively with JSTOR-Portico, and it could be applied to data from non-HathiTrust sources to create compatible datasets. This version of the extracted features vocabulary is designed as a linked data standard (JSON-LD).

Data Stats

# of volumes represented	17,123,746
# of pages represented	6,221,631,336
# of tokens represented	2,906,819,723,689
# files derived from in-copyright volumes	10,550,952
# pages derived from in-copyright volumes	3,743,561,467
# tokens derived from in-copyright volumes	1,709,172,692,891
# files derived from public domain & Creative Commons volumes	6,572,794
# pages derived from public domain & Creative Commons volumes	2,478,069,869
# tokens derived from public domain & Creative Commons volumes	1,197,647,030,798

Downloading the files

See directions for Extracted Features v.2.0

Attribution

Jacob Jett, Boris Capitanu, Deren Kudeki, Timothy Cole, Yuerong Hu, Peter Organisciak, Ted Underwood, Eleanor Dickson Koehl, Ryan Dubnicek, J. Stephen Downie (2020). *The HathiTrust Research Center Extracted Features Dataset (2.0)*. HathiTrust Research Center. https://doi.org/10.13012/R2TE-C227

Files and file format

1 Extracted Features file per volume

- There is one Extracted Features 2.0 file per volume from HathiTrust that has been processed into the dataset. (17.1 million volumes processed = 17.1 million Extracted Features files in the entire dataset.)
- The dataset is not continuously updated, and files are not created on request. There are volumes in HathiTrust for which an associated Extracted Features file does not exist.

Size of the entire dataset

- Each Extracted Features file is BZIP2 compressed.
- The entire compressed dataset is 4TB.
- For that reason, we advise against downloading the entire dataset unless you are prepared to store and work with that amount of data.

File sections

The main sections are the metadata block and the features block. For a basic overview, follow this introductory guide to the HTRC Extracted Features 2.0 format.

JSON-LD and the linked data format

Extracted Features 2.0 files are in JSON-LD format. The linked data integration is new to this version of HTRC Extracted Features, and incorporates semantics from the Schema.org web ontology with extensions of our devising.

Linked data is found primarily in the metadata section of the file, and it connects to ontologies and vocabularies like Schema.org, or to authorities databases such as Virtual International Authority File (VIAF) or Library of Congress linked data resources. When no entry was found in a 3rd-party database, a non-authoritative placeholder entity was created for the name in a named entity database maintained by HTRC.

Terminology

These terms found in the documentation may be unfamiliar.

Volume: A digitized item in the HathiTrust Digital Library, may be a book, an issue of a periodical, or several items bound together. Represents a physical item that would be pulled off a library shelf and digitized as one object.

Object: A key-value pair in the JSON file. Denoted by curly braces. E.g. {type:”Book”}

Node: A linked data concept, they are the individual elements that make up a JSON-LD document's graph (the subject-object-predicate of a linked data triple). Read more. They can be:

- A number or string
- Arrays (a list)
- Node objects (0 to more node properties)
- Arrays that list node objects

Term: A label for a particular node

Feature File Documentation

Repeating terms

Some terms in the Extracted Features files repeat, and their values depend on their context within the file. The repeating terms are described below.

id

Description: Used to uniquely identify the nodes composed by the Extracted Features file. Each id is a URI that allows a node to be externally referenced.

type

Description: Part of linked data standards. Wherever possible, we have used types that have already been defined in the Schema.org vocabulary. In places where this was not possible, we have either defined new types or extended Schema.org types through development of a sub-class.

name

Description: Used wherever Person and Organization object types are found in the metadata block. It identifies a Person or Organization in a human readable manner.

File description section

Each Extracted Features file opens with metadata describing the file itself.

@context

Description: Links to a context document written by HTRC, which maps terms in the Extracted Features 2.0 file. The context is written by HTRC.

Value: The URL identifying the context document. Always "https://worksets.htrc.illinois.edu/context/ef_context.jsonld"

schemaVersion

Description: The version of the extracted features schema.

Value: "3.0"

id

Description: See Extracted Features [v.2.0]#id

Value: The URL identifying the Extracted Features file. E.g., "https://data.analytics.hathitrust.org/extracted-features/20200210/ufl2.uf00103078_00001"

htid

Description: Every volume in HathiTrust is assigned a unique HathiTrust identifier (htid).

Value: The HathiTrust volume identifier. E.g. "ufl2.uf00103078_0000"

type

Description: See Extracted Features [v.2.0]#type

Value: "DataFeed" [Learn more]

publisher

Description: Describes HTRC, the publisher of the Extracted Features files. Consists of three parts: *id*, *type*, and *name*.

id

Description: See id

Value: “https://analytics.hathitrust.org”

Type

Description: See type

Value: “Organization”

Name

Description: See name

Value: “HathiTrust Research Center”

datePublished

Description: The date the Extracted Features file was created, in YYYYMMDD format [ISO 8601]

Value: E.g. “20200221”

Metadata section

Metadata primarily about the volume described by the Extracted Features file.

The metadata is received from HathiTrust in MARC (library catalog metadata) format. (Learn more about HathiTrust metadata.) It is then transformed to the Bibframe standard, which is a linked data standard. By shifting the metadata from MARC to Bibframe, we are able to improve the overall quality of the metadata by providing links to more contemporary authorities sources for metadata (such as VIAF, etc.). As a final step, the metadata is converted into the Extracted Features metadata section format.

If there would be no value for a field, then it is left out of the file. Therefore, you may not find all of the fields below in a given Extracted Features 2.0 file.

metadata

Description: Denotes the start of the metadata part of the Extracted Features file. The metadata describes the volume from HathiTrust represented by the Extracted Features file.

schemaVersion

Description: The schema version for the metadata block of the Extracted Features file.

Value: A URL linking to the schema. "https://schemas.hathitrust.org/Extracted Features_Schema_MetadataSubSchema_v_3.0"

id

Description: See id

Value: The Handle URL for the volume, which will point to the volume in the HathiTrust Digital Library. E.g. "http://hdl.handle.net/2027/mdp.39015062779023"

type

Description: See type

Value: Either “Book”, “PublicationVolume”, or “CreativeWork” depending on the value found in the Issuance field of the corresponding Bibframe record. When the Bibframe Issuance value is ‘mono’, the Book type is assigned. When the value is ‘serl’, the PublicationVolume type is assigned. In all other cases the CreativeWork type is assigned. The Bibframe Issuance is derived from a variety of fields in the MARC record, including Control Fields and Physical Description Fields.

dateCreated

Description: The date on which the metadata portion of the Extracted Features file is generated, in YYYYMMDD format [ISO 8601**]**

Value: E.g., “20200209”

title

Description: The title of the volume when the *type* (above) is “Book” or “CreativeWork”.

Value: Derived from the Bibframe title, which originates from the Title & Title-Related Fields in the MARC record.

alternateTitle

Description: An alternate title for a bibliographic entity described by the Extracted Features file.

Value: Derived from the Bibframe title where the *rdf:type* is “VariantTitle”. Originates from a variety of fields in the MARC record, including Main Entry Fields and Title & Title-Related Fields.

enumerationChronology

Description: Information regarding which volume, issue, and/or year the HathiTrust volume was published.

Value: Taken from the Bibframe enumerationAndChronology node, which is derived from a variety of fields in the MARC record, including Holdings Fields and Locally Added Fields.

publisher

Description: Information about the publisher of the volume described by the Extracted Features file. Includes *type*, *name,* and *id*. [Learn more]

type

Description: See type

Value: Either “Organization” or “Person”. Taken from the type node for the Bibframe provisionActivity’s agent.

id

Description: See id

Value: A URL identifying the publisher, such as a Handle URL, ORCID, etc.

Name

Description: See name

Value: Taken from the label node for the Bibframe provisionActivity’s agent. Derived from the Imprint field in the MARC record.

pubPlace

Description: Information about where the volume was first published. Includes *id* and *type*.[Learn more]

id

Description: See id

Value: Taken from the Bibframe Instance’s provisionActivity’s place *rdf:about* node, which are derived from the country codes in the MARC 008 field.

type

Description: See type

Value: “Place” [Learn more]

name

Description: See name

Value: Corresponds to information in the Imprint field in the MARC record.

pubDate

Description: The year in which that edition of the volume was first published.[Learn more]

Value: Taken from the Bibframe provisionActivity date, which is derived from the Imprint field in the MARC record.

genre

Description: Information about the volume's genre, as determined by the cataloger of the work. [Learn more]

Value: Taken from the Bibframe Work's genreForm node and corresponds to a controlled vocabulary, e.g., the MARC Genre Terms List. Values are derived from the Genre/Form Field in the MARC record.

category

Description: The volume's topic or topics. [Learn more]

Value: Derived from the Bibframe Work's ClassificationLcc node. Represents the natural language label for the Library of Congress Classification (LCC) value based upon the Library of Congress's LCC standard documentation.

language

Description: The cataloger-determined language or languages of the volume.

Value: Taken from the Bibframe Work's language's identifiedBy's value node, which is derived from the Language Code field in the MARC record.

accessRights

Description: The copyright status of the volume.

Value: Corresponds to attributes in HathiTrust's accessRights database. Derived from a HathiTrust-local MARC field (974r) that is added to bibliographic records as they are processed at HathiTrust.

lastRightsUpdateDate

Description: The most recent date the volume's copyright status was updated.

Value: Derived from a HathiTrust-local MARC field (974d) that is added to bibliographic records as they are processed at HathiTrust.

Contributor

Description: Contains information regarding the author(s), editor(s), or other agents involved in creating the volume. Consists of *id*, *type*, and *name*. [Learn more]

id

Description: See id

Value: A URL taken from the Bibframe agent that links to an authorities database (e.g., VIAF).

type

Description: See type

Value: Either the Person or Organization. Taken from the Bibframe agent type.

name

Description: The name of the person or organization who created the volume. See also name

Value: Taken from the Bibframe agent's label. Derived from a variety of fields in the MARC record, including Main Entry Fields, Title and Title-Related Fields, and Added Entry Fields.

typeOfResource

Description: The cataloger-determined resource type of the volume (e.g., text, image, etc.).

Value: The data value of this node is taken from the Bibframe Work's type node, which is derived from the MARC record's 008 Fixed-Length Data Elements Field.

sourceInstitution

Description: An array containing information about the institution that contributed the volume to HathiTrust. Always has a *type* node and a *name* node. [Learn more]

id

Description: See id

Value: A URL identifying the source institution

type

Description: See type

Value: Always "Organization"

name

Description: The name of the source institution. See also name

Value: Derived from a HathiTrust-local MARC field (974c) that is added to bibliographic records as they are processed at HathiTrust.

mainEntityOfPage

Description: An array of URLs linking to various metadata records describing the volume represented by the Extracted Features file. [Learn more]

Value: The array typically contains 3 URLs that point to the HathiTrust Bibliographic API: HathiTrust brief bibliographic record, HathiTrust full bibliographic record, and the HathiTrust catalog record.

oclc

Description: The OCLC number for the volume. An OCLC number is an identifier assigned to items as they are cataloged in a library.

Value: Derived from the MARC record's System Control Number field (035).

lcc

Description: The Library of Congress Classification number for the volume. An LCC number is a type of call number that would be used to locate an item on a library shelf.

Value: Derived from the MARC record's LC Classification Number field (053). E.g. PG7158.M76 E23

lccn

Description: The Library of Congress Control Number for the volume. An LCCN is a unique number that is assigned during cataloging.

Value: Derived from the MARC record's Library of Congress Control Number field (010).

issn

Description: The ISSN of the volume (when a journal). [Learn more]

Value: Taken from the Bibframe Instance's identifiedBy's Issn node, which is derived from the MARC record's ISSN field.

isbn

Description: The ISBN of the volume (when a book). [Learn more]

Value: Taken from the Bibframe Instance's identifiedBy's Isbn node, which is derived from the MARC record's ISBN field.

Features section

This portion of the Extracted Features file contains all of the unigram tokens (i.e. words), token counts, and other calculated or algorithmically-derived data from the HathiTrust volume.

The full text of the volume is tokenized using Stanford NLP, which includes part-of-speech detection. The tokens are tagged with part-of-speech tags using the tagsets from Stanford NLP.

After an initial section containing volume- and EF file-level metadata, this section is divided into JSON arrays for each page in the volume. Nested within those page-level arrays are arrays for the header, body, and footer of the page.

Fields that would be empty (i.e. if the algorithm does not detect content on a page) are given NULL values.

features

Description: Denotes the start of the features part of the Extracted Features file.

id

Description: See id

Value: The Handle URL identifying the HathiTrust volume

type

Description: See type

Value: “DataFeedItem”

schemaVersion

Description: The version of the features part of the Extracted Features document model.

Value: A URL linking to the schema. "https://schemas.hathitru…_FeaturesSubSchema_v_3.0"

dateCreated

Description: The date the data features were generated in in YYYYMMDD format. [ISO 8601] Note that this will be different from the dateCreated for both the metadata block and the overall Extracted Features file.

Value: E.g. “20200209”

pageCount

Description: The number of page scans in the volume, which is equivalent to the number of scanned images making up the digital object in the HathiTrust Digital Library.

Value: A number corresponding to the number of pages.

pages

Description: An array of individual page objects (one per page of the volume).

seq

Description: The sequence number of the page in the volume. Corresponds to the digital object, so that the first scan in the volume is “00000001”, which may be the cover, a title page, or something else.

Value: E.g. ““00000001" or “00000055”

version

Description: A hash of the page content used to compute the features for the page. Volumes in HathiTrust may be updated to improve scan or OCR quality or correct an issue, which would cause the text data to change, and, if features are reprocessed, a new hash would result.

Value: The alphanumeric hash.

calculatedLanguage

Description: The most probable language of the text on the page. Determined algorithmically, and specified by language codes.

Value: A code for the language. Will be null if no language detected, or if the language was not recognized by the algorithm.

tokenCount

Description: The total number of tokens detected on the page.

Value: A number representing the number of tokens.

lineCount

Description: The total number of lines of text detected on the page.

Value: A number representing the lines.

emptyLineCount

Description: The total number of empty lines on the page.

Value: A number representing the empty lines.

sentenceCount

Description: The total number of sentences detected on the page.

Value: A number representing the sentences.

header

Description: The header portion of the page.

tokenCount

Description: The total number of tokens detected in the header of the page.

Value: A number representing the tokens.

lineCount

Description: The total number of lines detected in the header of the page.

Value: A number representing the lines.

emptyLineCount

Description: The total number of empty lines detected in the header of the page.

Value: A number representing the empty lines.

sentenceCount

Description: The total number of sentences detected in the header of the page.

Value: A number representing the sentence count.

beginCharCount

Description: An array of the first non-White Space characters detected on lines in the header and their occurrance counts.

Value: The character and the number of times it occurs. E.g. “R:1”

endCharCount

Description: An array of the last non-White Space characters on detected lines in the header and their occurrance counts.

Value: The character (letter, number, or punctuation) and the number of times it occurs. E.g. “g:5”

tokenPosCount

Description: An array of individual tokens with their corresponding part of speech and occurrence counts in the header of the page. If the language is one that is recognized by the Stanford NLP parser ("ar", "zh", "en", "fr", "de", "es") StanfordNLP models for the language are used to tokenize the text and do part-of-speech tagging. Otherwise, a whitespace tokenizer is used.

Value: The token, its part of speech (represented by the Penn Tree Bank for English), and a number representing the number of times that token appeared in the header.

body

Description: The body portion of the page.

tokenCount

Description: The total number of tokens detected in the body of the page.

Value: A number representing the tokens.

lineCount

Description: The total number of lines detected in the body of the page.

Value: A number representing the lines.

emptyLineCount

Description: The total number of empty lines detected in the body of the page.

Value: A number representing the empty lines.

sentenceCount

Description: The total number of sentences detected in the body of the page.

Value: A number representing the sentence count.

beginCharCount

Description: An array of the first non-White Space characters detected on lines in the body and their occurrence counts.

Value: The character and the number of times it occurs. E.g. “R:1”

endCharCount

Description: An array of the last non-White Space characters detected on lines in the body and their occurrence counts.

Value: The character (letter, number, or punctuation) and the number of times it occurs. E.g. “g:5”

tokenPosCount

Description: An array of individual tokens with their corresponding part of speech and occurrence counts in the body of the page. If the language is one that is recognized by the Stanford NLP parser ("ar", "zh", "en", "fr", "de", "es") StanfordNLP models for the language are used to tokenize the text and do part-of-speech tagging. Otherwise, a whitespace tokenizer is used.

Value: The token, its part of speech (represented by the Penn Tree Bank for English), and a number representing the number of times that token appeared in the header.

capAlphaSeq

Description: The longest length of the alphabetical sequence of capital characters starting a line.

Value: A number representing the length.

footer

Description: The footer portion of the page.

tokenCount

Description: The total number of tokens detected in the footer of the page.

Value: A number representing the tokens.

lineCount

Description: The total number of lines detected in the footer of the page.

Value: A number representing the lines.

emptyLineCount

Description: The total number of empty lines detected in the footer of the page.

Value: A number representing the empty lines.

sentenceCount

Description: The total number of sentences detected in the footer of the page.

Value: A number representing the sentence count.

beginCharCount

Description: An array of the first non-White Space characters detected on lines in the footer and their occurrence counts.

Value: The character and the number of times it occurs. E.g. “R:1”

endCharCount

Description: An array of the last non-White Space characters on detected lines in the footer and their occurrence counts.

Value: The character (letter, number, or punctuation) and the number of times it occurs. E.g. “g:5”

tokenPosCount

Description: An array of individual tokens with their corresponding part of speech and occurrence counts in the footer of the page. If the language is one that is recognized by the Stanford NLP parser ("ar", "zh", "en", "fr", "de", "es") StanfordNLP models for the language are used to tokenize the text and do part-of-speech tagging. Otherwise, a whitespace tokenizer is used.

Value: The token, its part of speech (represented by the Penn Tree Bank for English), and a number representing the number of times that token appeared in the header.