Digging Deeper Reaching Further

Libraries Empowering Users to Mine the HathiTrust Digital Library Resources

Module 4.2 Performing Text Analysis: Basic Approaches with Python Lesson Plan

Further reading: go.illinois.edu/ddrf-resources

More advanced researchers will prefer to conduct text analysis outside of pre-built, off-the-shelf tools, opting instead for a toolkit of command line programs and custom code. This module introduces the concept of programming packages and provides hands-on experience with running Python code to analyze an Extracted Features file from the HTRC Extracted Features dataset.

Estimated time

50-65 minutes

Workshop audience

Librarians who want to develop their skillset for supporting researchers who want to engage in computational text analysis.

Learning goals

At the end of the module, the participants will be able to:

- Identify the needs of advanced text mining researchers in order to make skill-appropriate recommendations.
- Recognize text analysis methods in order to understand the kinds of research available in the field.
- Successfully interact with a pre-defined textual dataset in order to gain experience with programming skills for data-driven research.

Skills

Upon completion of the module, participants should be able to obtain the following skills:

- Install a Python library using Pip
- Run a Python script to work with an HTRC Extracted Features file

Prerequisites for participants

Ideally, participants:

- Have been introduced to the HTRC, or have completed Module 1
- Have used the command line, or have completed Module 2 Lesson 2
- Are acquainted with the programming language Python or, or have completed Module 3

Session outline

- Introduction to toolkit for do-it-yourself text analysis
- Overview of package managers and installing libraries/packages
- Introduction to HTRC Extracted Features
- **Activity:** Install a Python library and run a script to view most-used adjectives in a set of volumes
- Introduction to exploratory data analysis
- **Activity:** Install the HTRC Feature Reader and run Python script to view the word count in a volume based on its Extracted Features file
- Advanced text analysis with the HTRC Extracted Features example
- Discussion of the librarian's role in supporting text analysis research

Getting ready

Workshop participants will need:

- Access to a computer, the Internet, and a web browser
- Access to PythonAnywhere
- The following files in PythonAnywhere:
    - top_adjectives.py
    - word_count.py
    - mdp.49015002221860.json.bz2
    - mdp.49015002221878.json.bz2
    - mdp.49015002221886.json.bz2

    - miua.4925052,1928,001.json.bz2
    - miua.4925383,1934,001.json.bz2
    - mdp.49015002203033.json.bz2
    - mdp.49015002203140.json.bz2
    - mdp.49015002203157.json.bz2
    - mdp.49015002203215.json.bz2
    - mdp.49015002203223.json.bz2
    - mdp.49015002203231.json.bz2
    - mdp.49015002203249.json.bz2
    - mdp.49015002203272.json.bz2
    - mdp.49015002203405.json.bz2
    - mdp.49015002221761.json.bz2
    - mdp.49015002221779.json.bz2
    - mdp.49015002221787.json.bz2
    - mdp.49015002221811.json.bz2
    - mdp.49015002221829.json.bz2
    - mdp.49015002221837.json.bz2
    - mdp.49015002221845.json.bz2
- HTRC Feature Reader Python library installed to PythonAnywhere

Key concepts

- **Natural Language Processing (NLP):** Using computers to understand the meaning, relationships, and semantics within human-language text.
- **Named entity extraction:** Using computers to locate and classify named entities (such as the names of persons, organizations, and locations) in text.
- **Stylometry:** The application of the study of linguistic style. It is often used to determine authorship to anonymous or disputed texts.
- **Sentiment analysis:** Using computers to systematically identify attitudes or emotions present in text.
- **Machine learning:** A process that gives computers the ability to learn without being explicitly programmed. Machine learning is based on researchers constructing and using algorithms that can learn from and make predictions on data. It can either be unsupervised (with minimal human intervention) or supervised (with more human intervention).

- **Topic modeling**: A method of using statistical models for discovering the abstract "topics" that occur in a collection of documents.
- **Naïve Bayes classification:** A method based on Bayes' Theorem from statistics that uses machine learning to classify texts based on information present in the texts of each class.
- **Functions:** Reusable code blocks that perform an action.
- **Libraries/packages:** Collections of functions that can be implemented in a script or program.
- **Package Manager:** A tool that facilitates the download and installation of programming packages.
- **Exploratory data analysis:** An approach for familiarizing oneself with a dataset before analyzing it that often involves visualizations, including visualizations of raw counts and simple statistics, or comparative visualizations.

Key tools/platforms

- **Python:** A programming language that is good for working with data. Python has high-level data structures, is interpretive in nature, and has a relatively simply syntax.
- **pip:** Package manager for Python (alternatives: Homebrew, Conda).
- **R:** A programming language optimized for (statistical) data analysis.
- **HTRC Extracted Features:** A downloadable dataset of text data and metadata extracted and abstracted from volumes in the HathiTrust Digital Library.
- **HTRC Feature Reader:** Python library for working with HTRC Extracted Features.
- **pyplot:** Visualization function in the Python data science package, Pandas.

Key points

Key approaches to text analysis	• Among others, there are 2 key approaches to text analysis: natural language processing and machine learning • Natural language processing is the use of computers to understand the meaning, relationships, and semantics within human-language text. It includes named entity extraction, sentiment analysis, and stylometry. In many, but not all, cases, the researcher will require full text.

	• Machine learning is training computers to recognize patterns in text, and it can be supervised or unsupervised. It includes topic modeling and Naïve Bayes classification.
Activity: match project to method	• Participants match each of the research examples from Module 1 with a broad text analysis area and specific method. • *Goal:* Reinforce understanding the kinds of research questions that particular text analysis methods are suited to answer.
HTRC Extracted Features dataset	• A dataset of JSON files, one for each volume in the HTDL • The files contain metadata, including bibliographic metadata and computationally-derived metadata, such as word and line counts • They also include part-of-speech tagged token counts at the page-level
Do-it-yourself text analysis	• Some researchers will not be satisfied with pre-built, off-the-shelf tools. • They will want more control over the process via do-it-yourself tools
The text analysis toolkit	• The toolkit more advanced researchers will use depends on individual preferences • The researcher will likely need an understanding of statistics, and they may collaborate with other experts • The toolkit will consist of command line tools and programming languages • MALLET and Stanford NLP are common command line tools for text analysis • R and Python are common programming languages for text analysis
Programming concepts of modules, packages, and libraries	• Programming packages and libraries are collections of reusable code blocks; Packages are made up of modules • Packages for text analysis may facilitate tasks such preparing, reading or loading, and analyzing text with preset routines.

	• Packages are installed using a “package manager” which are command line tools that help make sure the packages are installed correctly
Activity: Install a Python library and run a script to view most-used adjectives in a set of volumes	• Using PythonAnywhere, instructors will guide participants through the process of installing the HTRC Feature Reader Python library and run a Python script to create a list of the most-used adjectives and the number of times they occur in a set of volumes in a workset. • *Goal:* Gain exposure to programming concepts, understand how counts of features can reveal information about text, practice basic text analysis.
Exploratory data analysis	• It is often difficult to grasp the contents of a dataset—its scope, range, and potential errors—from reading files alone. • Exploratory data analysis is the process by which one familiarizes themselves with a dataset before analysis • Often exploration involves visualization to make it easier to understand the data.
Activity: Visualize word count in an HTRC Extracted Features file	• Using a Python script, plot raw counts in an HTRC Extracted Features file • Visualize word count over a single volume • *Goal:* Develop comfortability with how basic text analysis can be aided by graphing data.
Advanced text analysis example	• Ted Underwood completed a text analysis project that used the HTRC Extracted Features dataset to classify volumes in the HTRC by genre • This work is an example of what can be done using the data fields in the Extracted Features and also of supervised machine learning

	• Ted released his derived dataset at the end of the project and it's available for others to use in their own analysis projects
Creativity Boom case study	• On his limited corpus of only pages containing the forms of "creativ*", Sam performed topic modeling • That way he ended up with the themes around the concept of creativity in the literature. • He then mapped the topics over time to see how their usage changed through the twentieth century.
Discussion	• In what ways can librarians support advanced text analysis research? • What additional skills would you need to learn in order to do so? • *Goal:* Encourage librarians to consider how they might apply what they have learned in the workshop.

Additional Tips for Instructors

- **Recommend participants NOT to use Internet Explorer for the web-based activities and choose an alternative browser such as Chrome or Firefox.** Participants using IE may encounter some issues with some of the activities.
- When demonstrating activities in web browsers, instructors may use "Ctrl" and "+" ("Command" and "+" on Macs) to enlarge the content on the screen. It can be quite difficult to see things from the back of the room! Use "Ctrl" and "-" ("Command" and "-" on Macs) to zoom back out when you need to demonstrate other things in regular size.