



MedTator: A Lightweight Interactive Multi-Document Annotation Tool

S36: Systems Demonstrations - Improving Workflow:
Informatics Tools No Researcher Should Be Without

Huan He, PhD*

Sunyang Fu, PhD, Liwei Wang, MD, PhD, Andrew Wen, MS, Sijia Liu, PhD, and Hongfang Liu, PhD †

Department of AI and Informatics, Mayo Clinic, Rochester, MN, USA

* He.Huan@mayo.edu

† Liu.Hongfang@mayo.edu

Disclosure

- I have no relevant relationships with commercial interests to disclose.

Learning Objectives

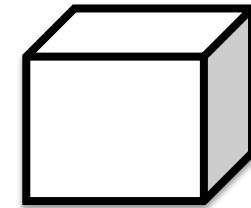
- Learn the serverless architecture for tool development
- Learn how to use MedTator for corpus annotation

Background

“Hi, I have wonderful data, could you _____ with AI?”



“Data”

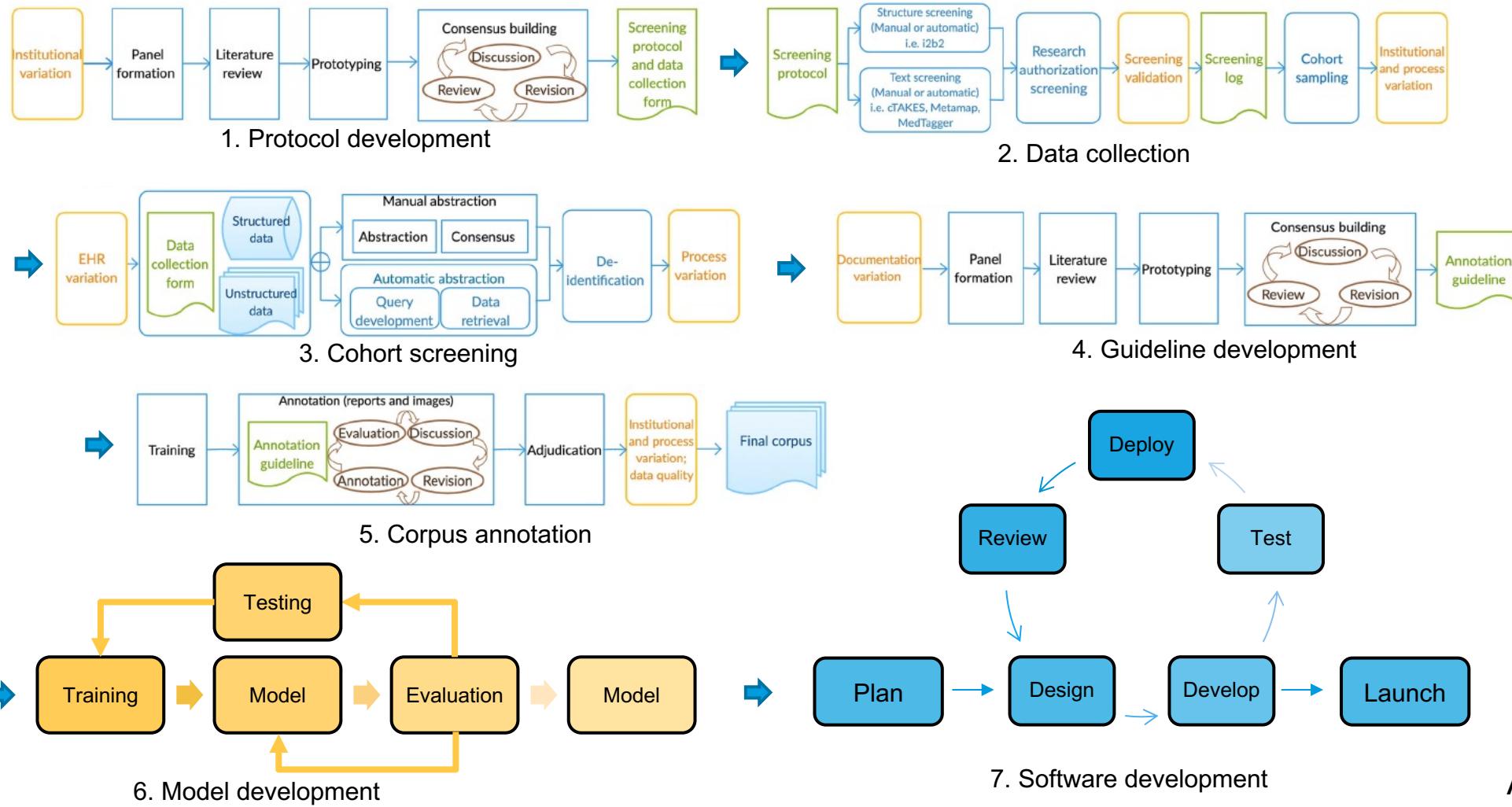


A fancy system

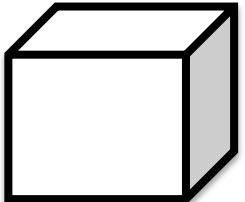
Adventure starts here



“Data”



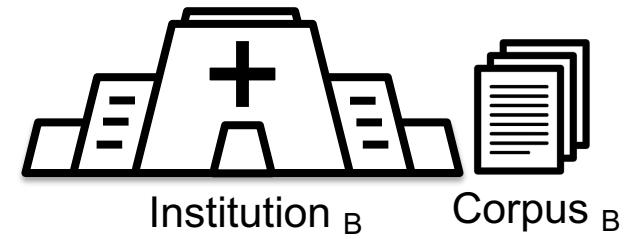
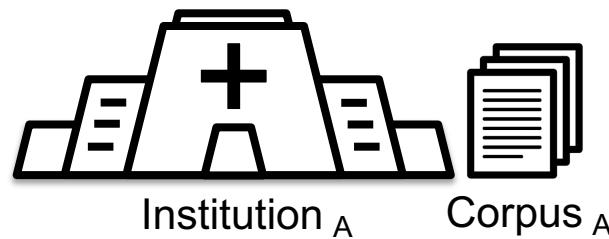
Annotated
Corpus



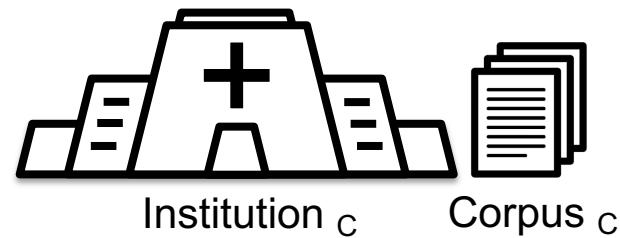
A fancy system

To conduct multi-site annotation

- Data security
- Corpus management
- Multi-site collaboration



A text annotation tool is needed!



Text Annotation Tools

<https://github.com/mariananeves/annotation-tools>

The screenshot shows the MAE interface with a table of events. Each event has columns for ID, TIME/EX, SIGNAL, AI/INK, SLINK, and TLINK. The table includes rows for various news items, such as N.V. DSM's net income and the Dutch chemical group's performance.

The screenshot shows the YEDDA-v1.0 Annotator interface. It features a main text area with annotations and a sidebar containing a 'Shortcuts map Labels' section with categories like Artificial, Event, Fin-Concept, Location, Organization, Person, Sector, and Other. A cursor is shown over the word 'OCCURRE...'.

The screenshot shows the Anafora interface. It displays a text document with annotations and a sidebar with sections like 'Patient Medical History' and 'Medications'. Annotations are highlighted in green and blue.

The screenshot shows the brat interface. It displays a text document with annotations and a sidebar with sections like 'Text', 'Annotations', 'Interventions', and 'Properties'. Annotations are highlighted in red and green.

The screenshot shows the INCEPTION interface. It lists numbered sentences from 18 to 23, each containing entities like PER, OTH, LOC, and PER. The interface includes a sidebar for active learning and a history section.

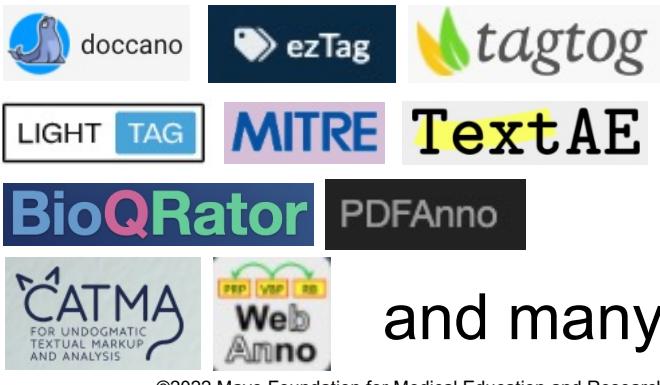
The screenshot shows the Prodigy interface. It displays a list of entities from the dataset 'prodigy_demo_ner_manual' and includes a progress bar at the bottom.

The screenshot shows the PubTator interface. It displays a search result for 'ESR1 breast cancer' with sections for 'Group / Sort', 'BioConcepts', and 'Text with highlighted annotations'.

The screenshot shows the TeamTat interface. It includes a 'RESULTS' section with a diagram of gene regulation, a 'MATERIALS AND METHODS' section with a table, and a 'REFERENCES' section with a list of publications.

The screenshot shows the FLAT interface. It features a 'Annotation Editor' window with a preview of a sentence and a 'Tag segment (tag span)' field. The background shows a large text document with various annotations.

The screenshot shows the LabelStudio interface. It features a 'Labeling controls' window with a 'Labels' section and a 'Segmentation (bounding box)' section. The background shows a detailed annotation interface with various components labeled.



and many!

The Installation of Existing Tools

1. Prepare a server and get permissions
2. Download the package of an annotation tool
3. Read through README, instruction of installation
4. Install the prerequisites
 - Oh, something wrong, version conflict, not found, exception, error ...
5. Install the tool and configure it
 - Oh, something wrong but different error
6. Finally, tool is installed but how to make it work?
7. Read through manual and try again
 - I did what they said in the manual, but it still didn't work, why?

The Architecture of Existing Tools

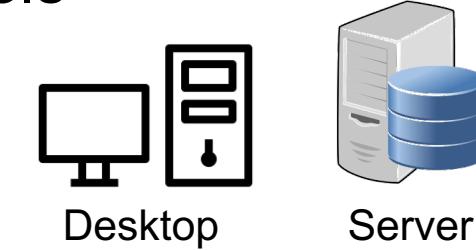
- Standalone tools

- MAE
- AGTK
- Analec
- @Note
- BioAnnotate
- Callisto
- Ellogon
- Glozz
- MMAX2
- NOMAD
- RAD



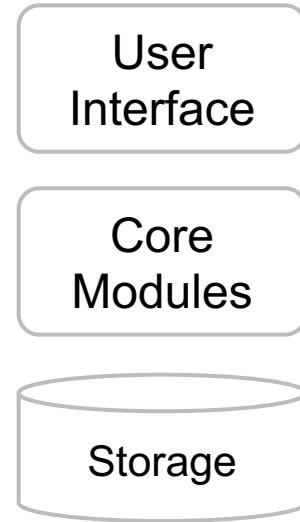
- Web-based tools

- Anafora
- eHost
- YEDDA
- BioQRator
- brat
- Catma
- FLAT
- WebAnno
- INCEpTION
- MAT
- TextAE



System Architecture Perspective

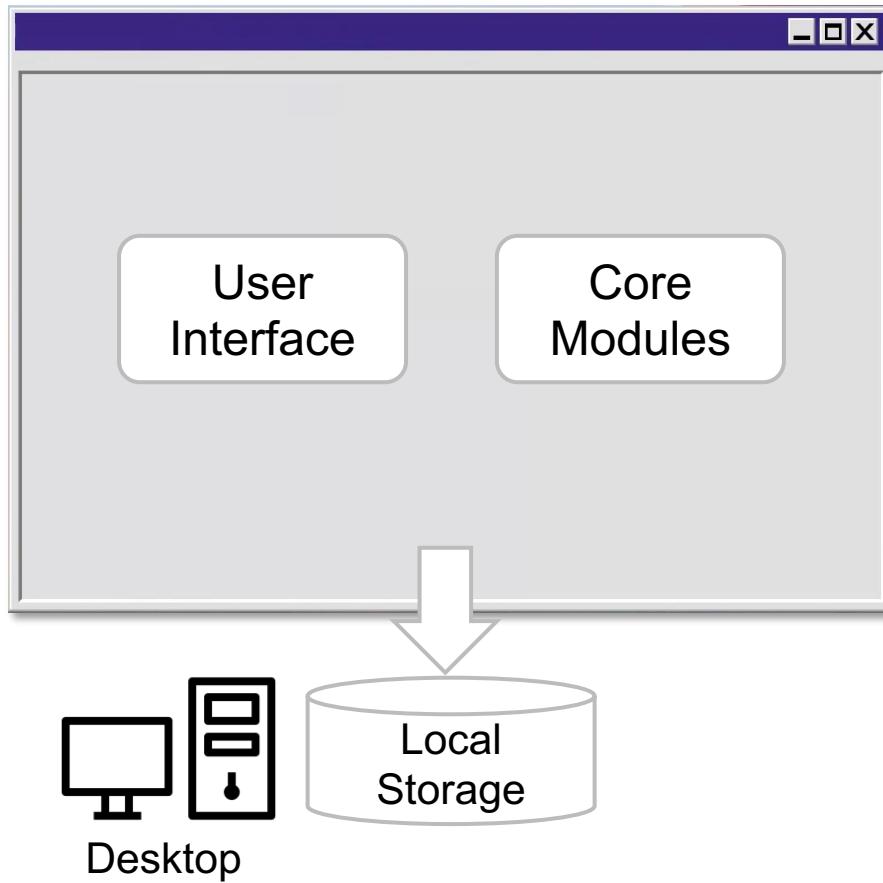
- Input
 - How the data enter system?
- Process
 - What to do with the entered data?
- Output
 - Where to save the processed data?



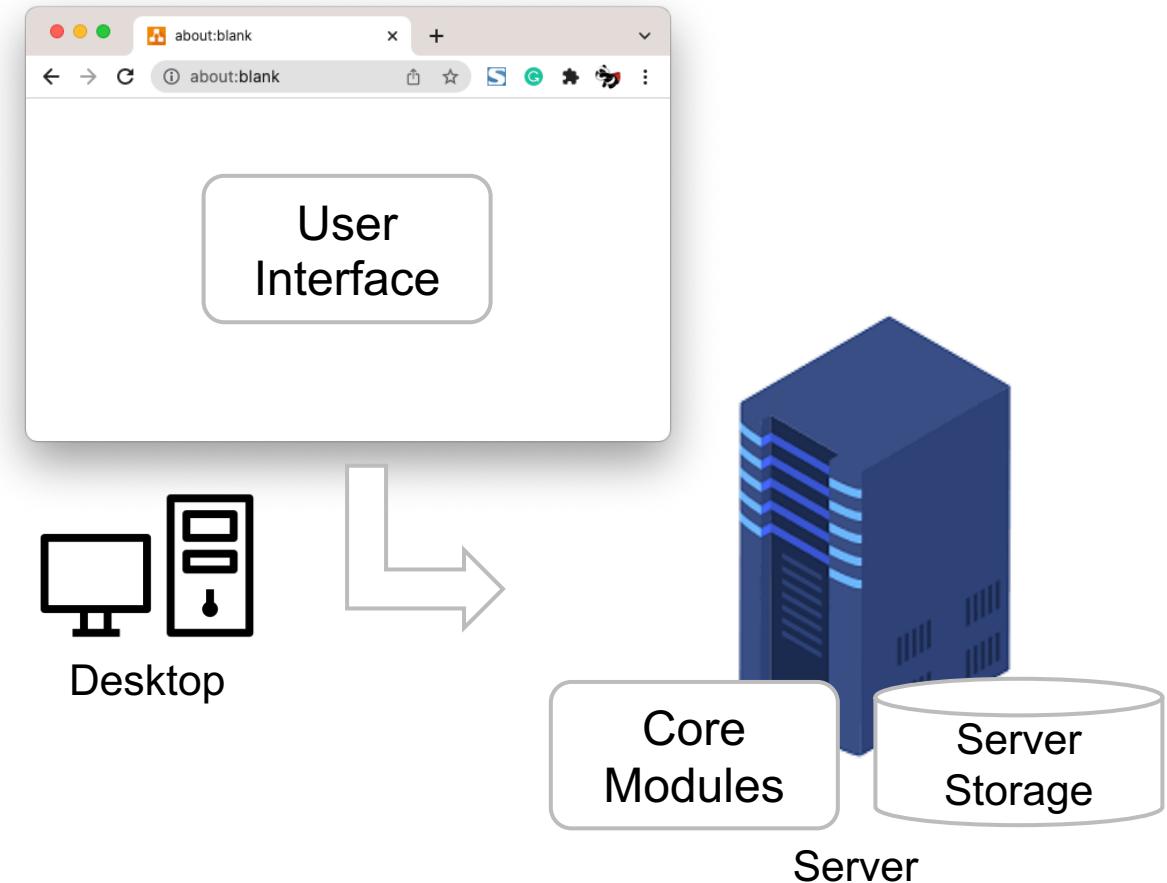
Text annotation Tool

The Architecture of Existing Tools

- Standalone tools

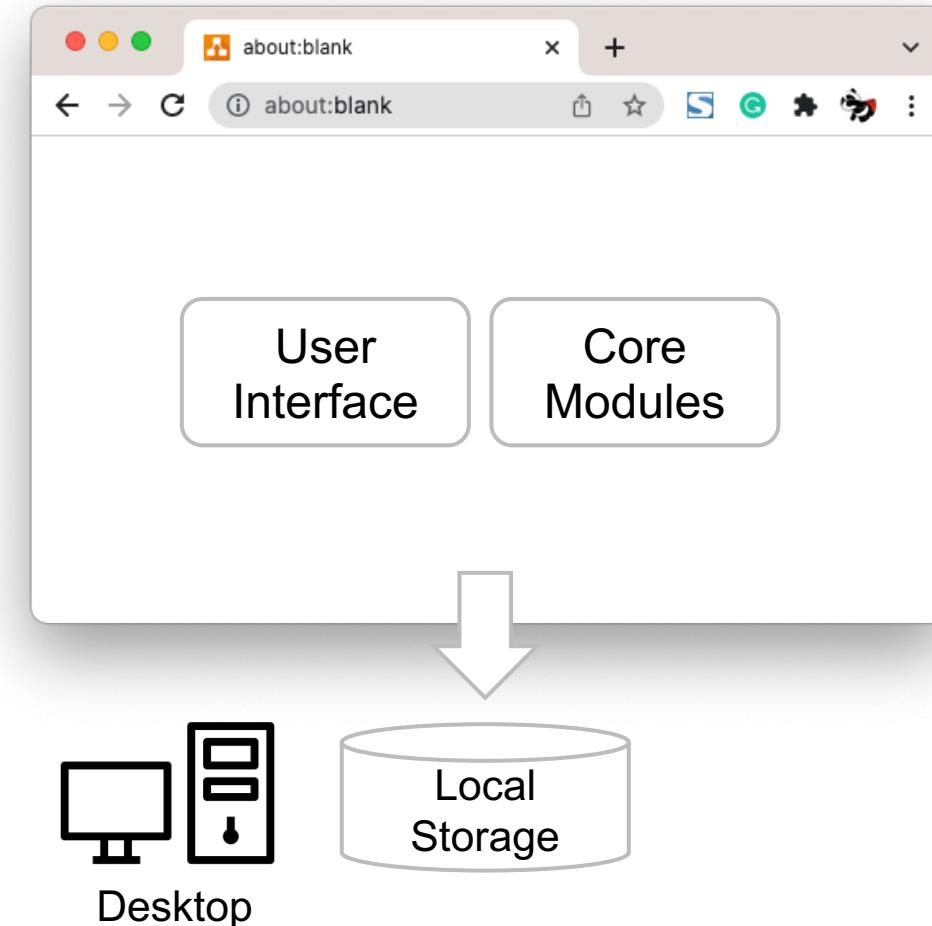


- Web-based tools



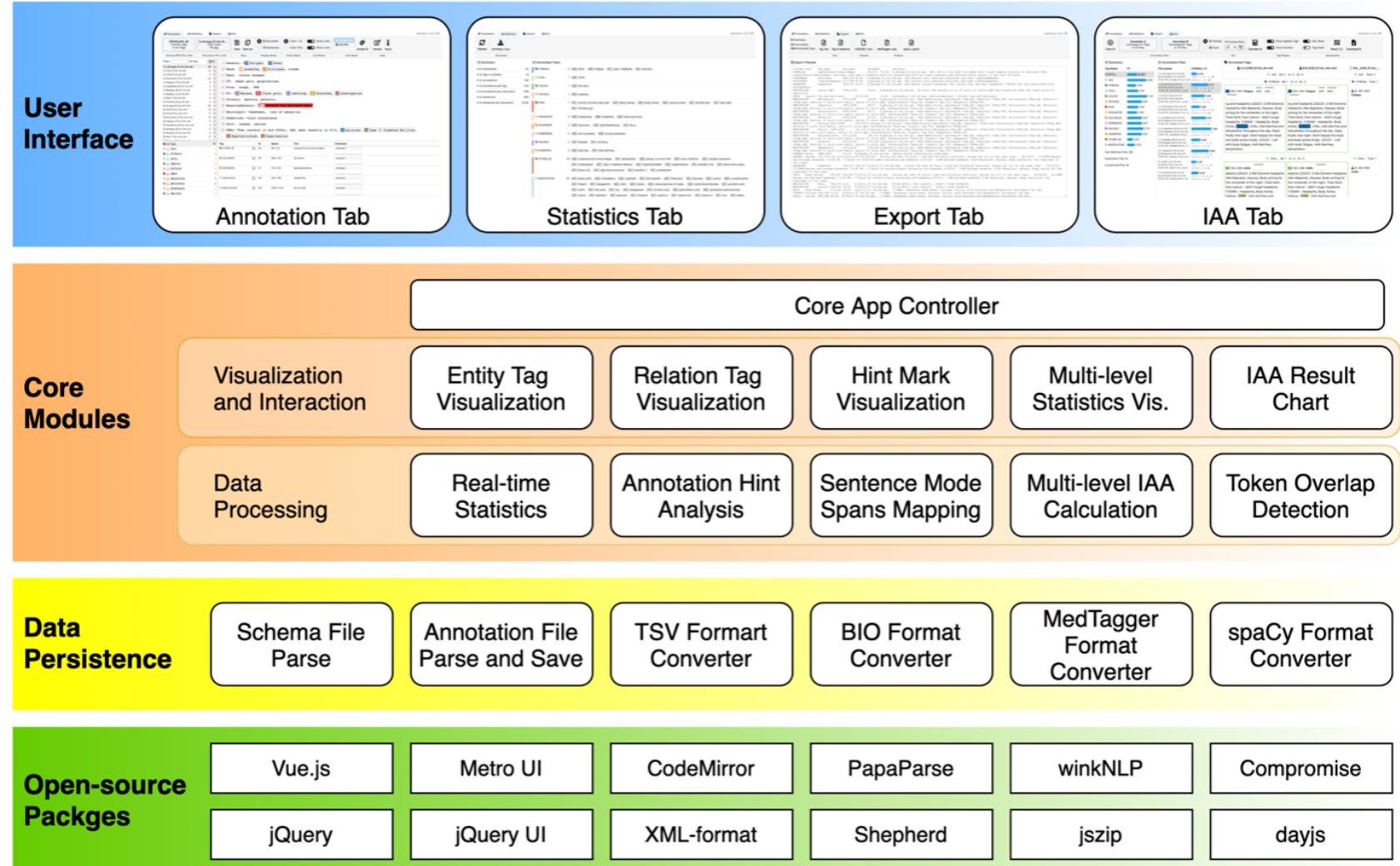
MedTator - Serverless Architecture

- Similar to standalone tools:
 - Data is saved in local storage
 - UI and core run together
- Similar to web-based tools:
 - UI runs in web browser
- Advantages
 - No need to install



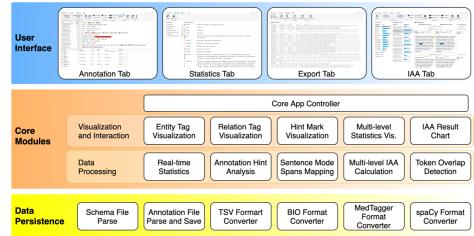
Functionalities

- Text annotation
- Statistics
- Adjudication
- Export



Implementation

Single HTML file



index.html



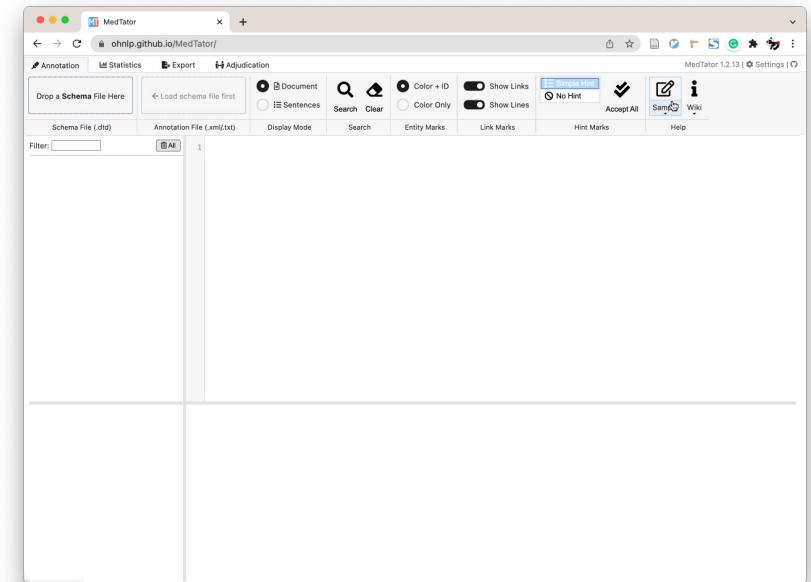
GitHub Pages

JS libraries



 **UNPKG**
 **JSDELIVR**  **CLOUDFLARE®**
Content Delivery Network (CDN)

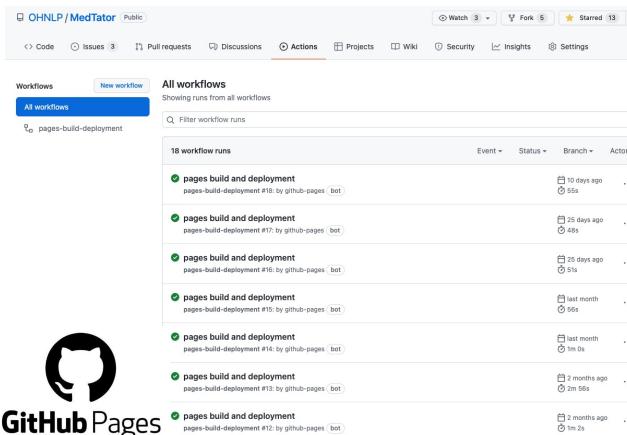
Web Browser



Chromium-
based browser

Advantages

- Researchers are not concerned with capacity planning, management, maintenance, or scaling of virtual machines, or physical servers.
- **Free** public services: GitHub Pages, GitLab Pages, BitBucket Pages...
- Easy to deploy, “install”, and update



`git push` to GitHub Pages to deploy on free domain name
<https://ohnlp.github.io/MedTator/>

The screenshot shows the MedTator application interface. On the left, a clinical note for a patient with COVID-19 symptoms is displayed. On the right, a table of annotations is shown, with one row highlighted. The table columns include Tag, ID, Span, Text, and Attributes.

Tag	ID	Span	Text	Attributes
1 PYREXA	P10	7-15	feverish	certainly comment
2 HEADACHE				
3 PAIN				
4 PYREXA	0	39-47	Soreness	certainly Positive comment
5 FATIGUE				
6 DIZZINESS				
7 NAUSEA				
8 OTHER_AE	01	52-62	tenderness	certainly Positive comment
9 VAX	VO	198-204	BNT162B2	comment
10 MEDICATION	M0	666-690	acetamin hydrochloride	comment

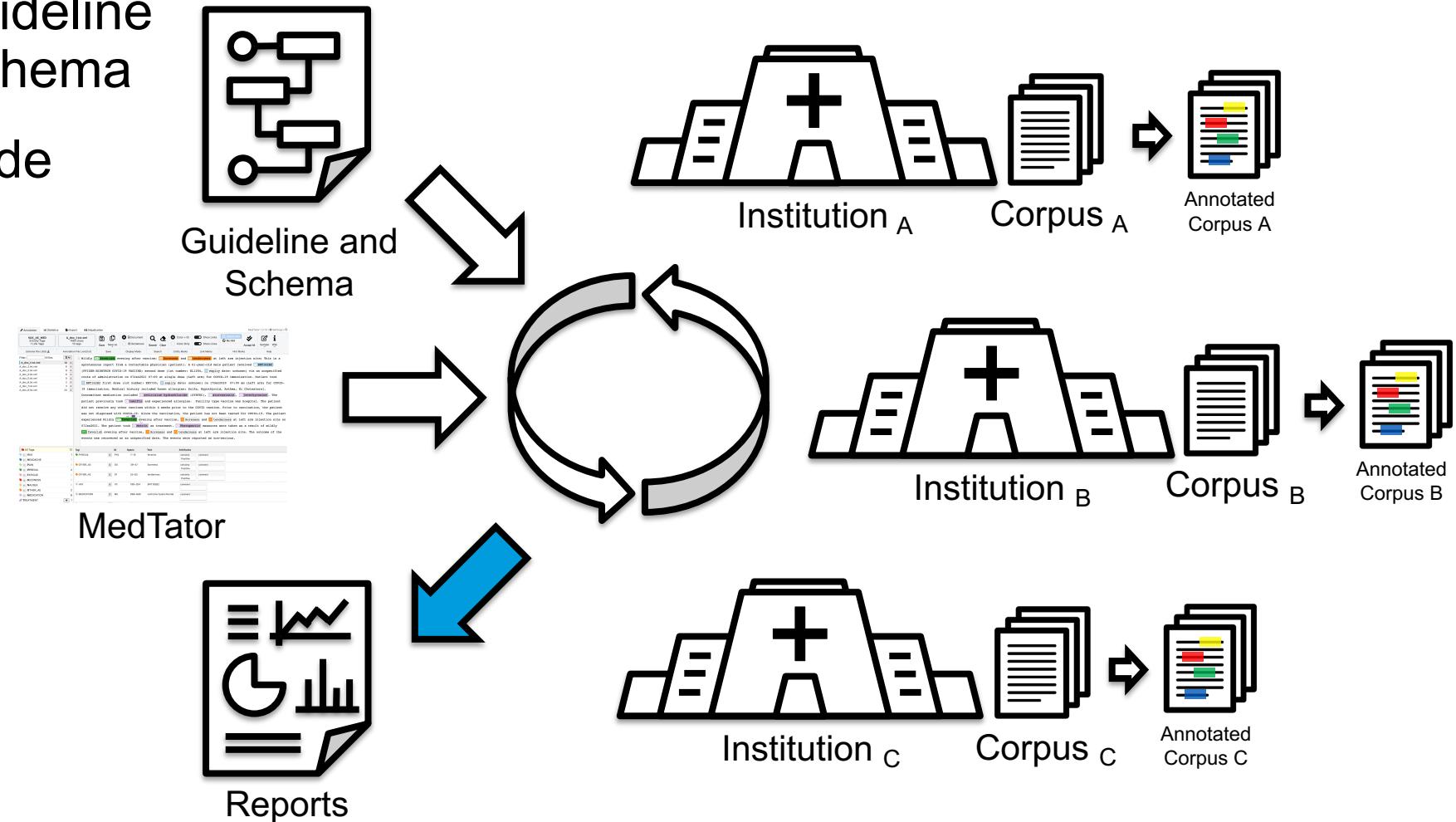
Global access with fast public CDN
No need to install
(Or run as a local static webpage)

The screenshot shows the GitHub Releases interface for the 'MedTator' repository. It lists two releases: 'Release version 1.2.13' (last updated 11 days ago) and 'Release version 1.2.10' (last updated 25 days ago). Each release page includes a summary, download links for source code and standalone.html, and a 'Compare' button.

Release new version online

Advantages - Federated annotation

- Only share the guideline and annotation schema
- No centralized node
- No server cost
- No PHI data transfer
- Same annotation user experience



Limitations - File management in HTML5

- File system access API
 - Not supported by all browsers

The screenshot shows the MDN Web Docs page for the File System Access API. The page title is "File System Access API". A note in the main content area states: "Secure context: This feature is available only in [secure contexts](#) (HTTPS), in some or all [supporting browsers](#)". Below this, a paragraph explains: "The File System Access API allows read, write and file management capabilities." The left sidebar contains "RELATED TOPICS" for the File System Access API, including sections for "Interfaces" (FileSystemHandle, FileSystemFileHandle, FileSystemDirectoryHandle, FileSystemWritableFileStream) and "Methods" (window.showOpenFilePicker, window.showSaveFilePicker, window.showDirectoryPicker, DataTransferItem.getAsFileSystem). The right sidebar includes "IN THIS ARTICLE" links for Concepts and Usage, Interfaces, Examples, Specifications, and Browser compatibility.

Browser compatibility

[Report problems with this compatibility data on GitHub](#)

	Chrome	Edge	Firefox	Internet Explorer	Opera	Safari	WebView Android	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet
<code>FileSystemHandle</code> ⚠	✓ 86	✓ 86	✗ No	✗ No	✓ 72	✓ 15.2	✗ No	✓ 86	✗ No	✗ No	✓ 15.2	✓ 14.0
<code>isSameEntry</code> ⚠	✓ 86	✓ 86	✗ No	✗ No	✓ 72	✓ 15.2	✗ No	✓ 86	✗ No	✗ No	✓ 15.2	✓ 14.0
<code>kind</code> ⚠	✓ 86	✓ 86	✗ No	✗ No	✓ 72	✓ 15.2	✗ No	✓ 86	✗ No	✗ No	✓ 15.2	✓ 14.0
<code>name</code> ⚠	✓ 86	✓ 86	✗ No	✗ No	✓ 72	✓ 15.2	✗ No	✓ 86	✗ No	✗ No	✓ 15.2	✓ 14.0
<code>queryPermission</code> ⚠ ⚠	✓ 86	✓ 86	✗ No	✗ No	✓ 72	✗ No	✗ No	✓ 86	✗ No	✗ No	✗ No	✓ 14.0
<code>requestPermission</code> ⚠ ⚠	✓ 86	✓ 86	✗ No	✗ No	✓ 72	✗ No	✗ No	✓ 86	✗ No	✗ No	✗ No	✓ 14.0

✓ Full support

✗ No support

⚠ Experimental. Expect behavior to change in the future.

⚠ Non-standard. Check cross-browser support before using.

Limitations - Functionality (for science)

- Limited support in scientific computing libraries for advanced features

	JavaScript	Python
Computation	NumJS	NumPy
	Danfo.js	Pandas
	std.js	SciPy, statsmodels
Machine Learning / NLP	tensorflow.js compromise, winkNLP	scikit-learn, tensorflow spaCy, flair
Data Visualization	D3.js, p5.js ECharts Plotly	Matplotlib, seaborn

Demonstration

- Live demo with sample datasets:
<https://ohnlp.github.io/MedTator/>

The screenshot shows the MedTator 1.2.0 interface. At the top, there's a navigation bar with tabs for Annotation, Statistics, Export, and Adjudication. Below the tabs, there are buttons for Save, Save as, Document, Sentences, Color + ID, Show Links, Simple Hint (which is selected), Color Only, Show Lines, Accept All, Sample, and Wiki. The main area displays a document with several annotated words highlighted in orange. The sidebar on the left shows a list of files and a tag hierarchy. The tag hierarchy includes VAX, HEADACHE, PAIN, PYREXIA, FATIGUE, and DIZZINESS, with VAX having one child node, HEADACHE having two, and so on.

Simple Hint
No Hint

Accept All

Sample Wiki

MedTator 1.2.0 | Settings

VAX_AE_MED
9 Entity Tags
1 Link Tags

A_doc_1.txt.xml
2327 chars
9 tags

Save Save as Document Sentences Color + ID Show Links Simple Hint No Hint Accept All Sample Wiki

Schema File (.dtd) Annotation File (.xml/.txt) Save Display Mode Search Entity Marks Link Marks Hint Marks Help

Filter: 12 files

A_doc_1.txt.xml 9
A_doc_2.txt.xml 6
A_doc_3.txt.xml 8
A_doc_4.txt.xml 6
A_doc_5.txt.xml 12
A_doc_6.txt.xml 2
A_doc_7.txt.xml 2
A_doc_8.txt.xml 24
A_doc_9.txt.xml 2
A_doc_10.txt.xml 4
A_doc_11.txt.xml 2
A_doc_12.txt.xml 1

All Tags 9
VAX 1
HEADACHE 2
PAIN 0
PYREXIA 0
FATIGUE 0
DIZZINESS 2

Tag ID Spans Text Attributes

Tag	ID	Spans	Text	Attributes
OTHER_AE	O0	86~93	anxiety	comment
OTHER_AE	O1	94~105	nervousness	comment
DIZZINESS	D0	199~204	dizzy	comment

Thank you!



Email: Hongfang Liu (Liu.Hongfang@mayo.edu)
Huan He (He.Huan@mayo.edu)

MedTator: <https://ohnlp.github.io/MedTator/>

The screenshot shows the MedTator 1.2.0 interface with the following details:

- Annotation:** A tab bar with Annotation, Statistics, Export, and Adjudication.
- File Information:** Shows "VAX_AE_MED" with 9 Entity Tags and 1 Link Tags, and "A.doc_1.txt.xml" with 2327 chars and 9 tags.
- Tool Buttons:** Save, Save as, Document, Sentences, Search, Clear, Color + ID, Color Only, Show Links, Show Lines, Simple Hint (selected), No Hint, Accept All, Sample, and Wiki.
- File List:** A list of 12 XML files from "A.doc_1.txt.xml" to "A.doc_12.txt.xml".
- Text Preview:** A large text area showing a patient's note about COVID-19 vaccination symptoms and a medical emergency.
- Entity Extraction Table:** A table showing extracted entities with columns: Tag, ID, Spans, Text, and Attributes.
- Tag List:** A sidebar showing a list of extracted tags: VAX, HEADACHE, PAIN, PYREXIA, FATIGUE, and DIZZINESS.

Tag	ID	Spans	Text	Attributes
OTHER_AE	O0	86~93	anxiety	comment
OTHER_AE	O1	94~105	nervousness	comment
DIZZINESS	D0	199~204	dizzy	comment