If you run out of room, try to reduce the size of your fonts and/or the size of your etc. Once you finish adding your content you can go back and format your text as formatting. Use the default font size even if your text extends beyond the bottom of the poster.

Adding content to the poster

The dotted lines on his poster template are guides. The horizontal and vertical guides Ruler and Guides

Zoom in and look at your images at 100%-200% magnification. If they look clear, they disproportionally.

You can add photos by dragging and dropping from your desktop, copy and paste, or Photos

To print your poster using our same-day professional printing service, go online to

60 inches wide

of your graphics. The font used for references can be smaller.

graphics. If there is a lot of empty space try to increase your font sizes and the size template for your poster presentation.

Before you start working on your poster and to

Click inside a section header to add its text.

For better zoom flexibility, use the zoom tab.

Distance measurement / authorship attribution

The frequency of the elements of a set of tokens associable with a social group (e.g. the class of words) counted in fixed-sized substrings throughout a document

Narrative time / authorial intent analysis

Applying delta analysis (e.g. Eden’s Delta, Connors, Martinsson Burns) to all of text to investigate similarities and identify possible authors of documents of anonymous authorship

TEI XML Viewer

Functionalities and services of the DH-LAB infrastructure:

- Inverted/OKI Data Repository
- Secondary GIS
- Batch data enrichment (i.e. external input of large amounts of data/metadata from file dialog or command line)
- OCR service
- Proofreading and transcription
- Semantic indexing and grid building from metadata and plaintext
- NLP (tokenization, lemmatization, NER, etc.) optimized to English and Hungarian
- TEI XML search
- TEI XML visualization

DH-LAB Proofreading / Transcription module enables users to process files entered in a data repository

- connect locally on-premises OCR results
- transcribers document from scratch (left-to-right and right-to-left) text orientation is defined at line level, and conclusively transcribing “mixed” documents is supported
- save transcribing data into separate modules
- HTR [Ottoman or other documents] with OCR (under development)

DH-LAB TEI XML Viewer and Search Engine

Product Owner and Product Manager: Tamás Kiss
Software development: Zoltán Kamász-Nagy (DevLead), Dóra Honvéd-Györnök, Anett Már

Rumi 1.1: Word counter and lemmatizer for Ottoman Turkish

Rumi 1.1 is a text analysis software optimized for the Ottoman language, which I have been developing since 2019. Essentially, Rumi is a lemmatizer and parameterizable word counter applicable to texts transcribed to the Latin script. Currently, Rumi 1.1 enables users to perform the following processes:

- Tokenization (i.e. splitting the text into words)
- Lemmatization of words (i.e. removing suffixes) featuring Turkic morphology
- Character normalization (i.e. eliminating differences between transcription standards)
- Word counting with the following parameters:
  - Individual document / various documents compiled
  - Number of most frequent words
  - Lemmatized / not lemmatized
  - Including / excluding stopwords
  - Absolute frequency / relative frequency
- "Supervised" word counting to count the elements of a predefined set of words within a document or documents with the following parameters:
  - In subunits (i.e. substrings) defined by word count
  - In subunits defined by their number within the document
    With subunits shifted by an adjustable number of words
- P-A-R analysis to count the number of words of Persian, Arabic and Turkic origin within a document
- Simple visualization to display word counter results
- Exporting data in CSV

Results with Rumi 1.1

Theme identification

A subunit’s relative frequency of lemmatized types (i.e. elements of the lexicon) compared with every subunit’s relative frequency of lemmatized types in other documents

DH-LAB TEI XML Viewer Examples

All versions view with a critical note and the occurrence of a word in specific versions displayed

DH-LAB TEI XML Viewer

Search and filter categories in the TEI XML Search Engine:

- Metadata
  - titles
  - People
  - Dates
  - Places
  - Institutions
  - Manuscript features
  - Language
- FullText
  - full-text versions
  - in specific text versions
  - Named entities
  - People

TEI XML Viewer is used to display the textual content and all added semantic, linguistic, textological, and critical information encoded into TEI XML. Rumi to make them easily explorable for human readers. It enables users to explore all versions of the same text of once or in parallel views, navigate OCR-ed / HTR-ed or transcribed documents in parallel views of the image layer and the text layer, and much more. The TEI XML Search Engine makes all unique features of TEI XML searchable of metadata, fullText, and entity level. Searching and filtering for persons (such as editors, addresses of letters, people mentioned in the text, etc.) or place names (such as place of publication, geographical locations featured in the text, etc.) make investigating trends and features easier across entire corpora. By named entity recognition the engine also enables users to search for people and place names that are not mentioned in the fullText by name. E.g. searching for Budapest will find “I went to the capital” if in the text “capital” refers to Budapest.

Key features of TEI XML Viewer:

- Basic search for records and files in linked GitHub repositories and data repositories
- Whole screen and split screen views
- One text version, all text versions, and simplified all text versions view
- Paginated and scrollable layout
- Annotations view
- Phen-view (i.e. printable critical edition with all textual and added information featured in footnotes)

Search Engine

Users to explore all versions of the same text at once or in parallel views, navigate OCR-ed / HTR-ed or transcribed documents in parallel views of the image layer and the text layer, and much more. The TEI XML Search Engine makes all unique features of TEI XML searchable of metadata, fullText, and entity level. Searching and filtering for persons (such as editors, addresses of letters, people mentioned in the text, etc.) or place names (such as place of publication, geographical locations featured in the text, etc.) make investigating trends and features easier across entire corpora. By named entity recognition the engine also enables users to search for people and place names that are not mentioned in the fullText by name. E.g. searching for Budapest will find “I went to the capital” if in the text “capital” refers to Budapest.

Key features of TEI XML Viewer:

- Basic search for records and files in linked GitHub repositories and data repositories
- Whole screen and split screen views
- One text version, all text versions, and simplified all text versions view
- Paginated and scrollable layout
- Annotations view
- Phen-view (i.e. printable critical edition with all textual and added information featured in footnotes)

Search Engine

Users to explore all versions of the same text at once or in parallel views, navigate OCR-ed / HTR-ed or transcribed documents in parallel views of the image layer and the text layer, and much more. The TEI XML Search Engine makes all unique features of TEI XML searchable of metadata, fullText, and entity level. Searching and filtering for persons (such as editors, addresses of letters, people mentioned in the text, etc.) or place names (such as place of publication, geographical locations featured in the text, etc.) make investigating trends and features easier across entire corpora. By named entity recognition the engine also enables users to search for people and place names that are not mentioned in the fullText by name. E.g. searching for Budapest will find “I went to the capital” if in the text “capital” refers to Budapest.

Key features of TEI XML Viewer:

- Basic search for records and files in linked GitHub repositories and data repositories
- Whole screen and split screen views
- One text version, all text versions, and simplified all text versions view
- Paginated and scrollable layout
- Annotations view
- Phen-view (i.e. printable critical edition with all textual and added information featured in footnotes)

Search Engine

Users to explore all versions of the same text at once or in parallel views, navigate OCR-ed / HTR-ed or transcribed documents in parallel views of the image layer and the text layer, and much more. The TEI XML Search Engine makes all unique features of TEI XML searchable of metadata, fullText, and entity level. Searching and filtering for persons (such as editors, addresses of letters, people mentioned in the text, etc.) or place names (such as place of publication, geographical locations featured in the text, etc.) make investigating trends and features easier across entire corpora. By named entity recognition the engine also enables users to search for people and place names that are not mentioned in the fullText by name. E.g. searching for Budapest will find “I went to the capital” if in the text “capital” refers to Budapest.

Key features of TEI XML Viewer:

- Basic search for records and files in linked GitHub repositories and data repositories
- Whole screen and split screen views
- One text version, all text versions, and simplified all text versions view
- Paginated and scrollable layout
- Annotations view
- Phen-view (i.e. printable critical edition with all textual and added information featured in footnotes)

Search Engine

Users to explore all versions of the same text at once or in parallel views, navigate OCR-ed / HTR-ed or transcribed documents in parallel views of the image layer and the text layer, and much more. The TEI XML Search Engine makes all unique features of TEI XML searchable of metadata, fullText, and entity level. Searching and filtering for persons (such as editors, addresses of letters, people mentioned in the text, etc.) or place names (such as place of publication, geographical locations featured in the text, etc.) make investigating trends and features easier across entire corpora. By named entity recognition the engine also enables users to search for people and place names that are not mentioned in the fullText by name. E.g. searching for Budapest will find “I went to the capital” if in the text “capital” refers to Budapest.