Visualization of LLM Annotated Documents

Teodor Valchev and Nikolay Paev

Artificial Intelligence and Language Technology Dept. , IICT-BAS
teodorvulchev@gmail.com and nikolay.paev@iict.bas.bg

1. Overview

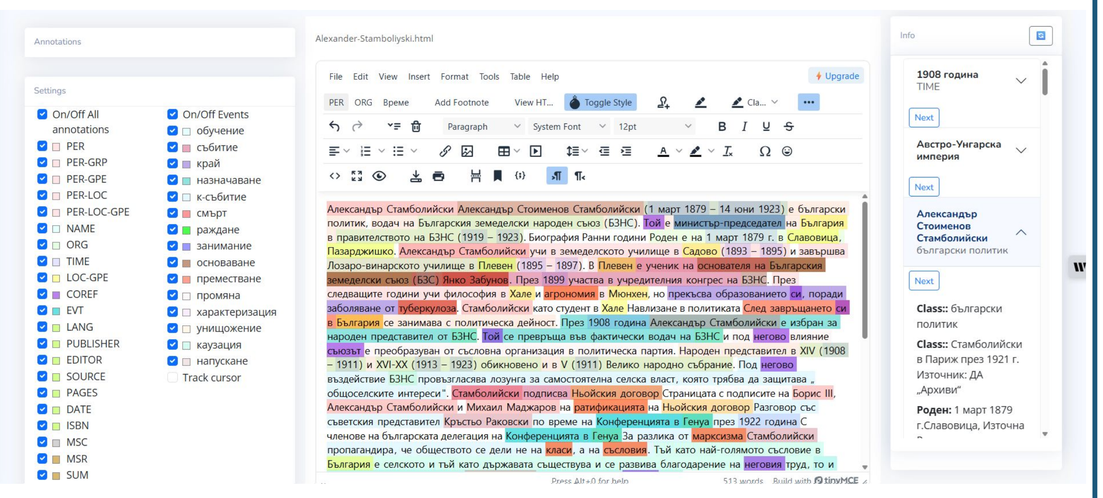
Motivation

Manual annotations remain essential for advancing Natural Language Processing.

Aim

- Supporting researchers in the area of Social Sciences and Humanities (SS & H) in doing their investigations.
- The system provides User Interface (UI) to semantically annotated documents, related to a knowledge graph representing knowledge from CLaDA-BG project.
- The system provides different annotation options: Automatic by LLMs, Manual mode, LLM assisted by LLMs.
- The UI is built over What You See Is What You Get (WYSIWYG) editor – TinyMCE, which is HTML-based text editor, extended for our needs.

2. WYSIWYG editor



- Left column: The left column displays the annotation semi-transparent coloring toggles
- Middle column: The center window is the interface of the extended TinyMCE editor.
- Right column: The right column shows information from the knowledge base for the entities in the document.

Coloring is handled with Cascading Style Sheets (CSS), but browsers allow only one rule of the same type at a time. To bypass this, dynamic CSS rules (for single and multiple classes) are generated in the browser as the document loads, limited to the class combinations present in the document, which prevents exponential growth.

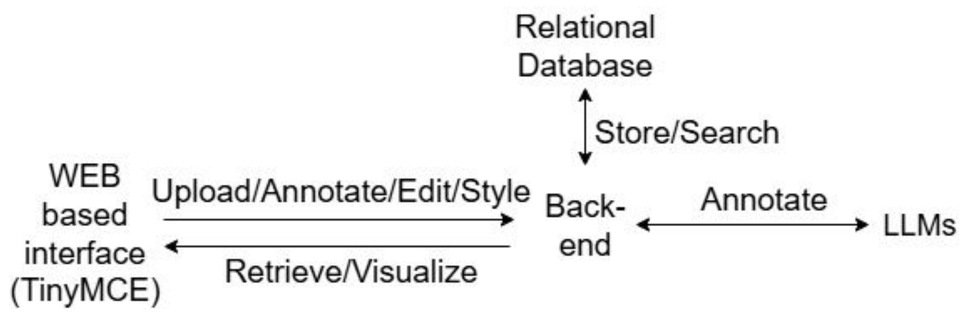
Documents are represented in TinyMCE as XHTML extended with a single new <tok> tag with multiple custom attributes, very similar to SpaCy.

3. Language models annotations

We are using pre-trained and later fine-tuned BERT and T5 models for initial NER and Event annotations.

The models are trained on the Bulgarian Event corpus, more details are available in *Bulgarian event extraction with llms, (*Simov et al.) at RANLP 2025.

4. Main workflow of the editor



1. The users uploads or creates a plain document which internally is presented as HTML.
2. The document is saved on the server and sent to LLM for automatic annotation.
3. After automatic annotation, the results are saved in the database.
4. After the document is processed, it is returned in cladaHTML to the UI, the user can edit, style, edit annotations, create a new annotation, etc.

5. Bulgaria-Centric Knowledge Graph

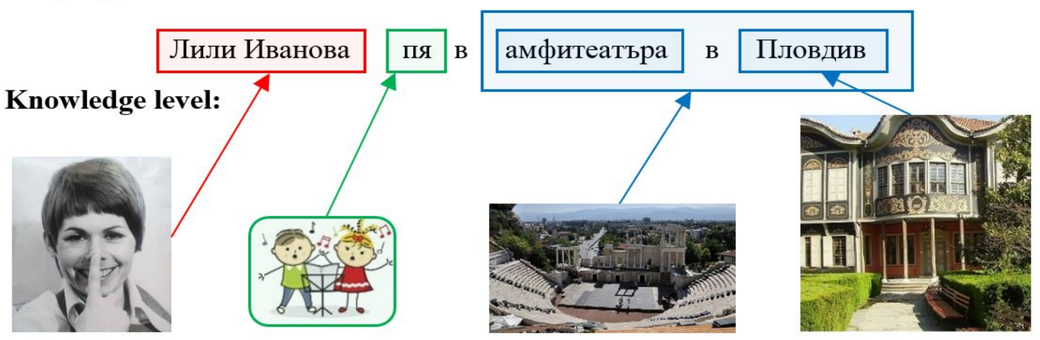
The integrated language resources and the mapping to the knowledge graph is basis for creation of many applications:

- Knowledge extraction from text
- Semantic annotation of documents
- Language access to the knowledge graph
- Integration of the scientific data of CLaDA-BG into the Knowledge Graph
- Reasoning
- Others

Language level:

Лили Иванова | пя | в | амфитеатъра | в | Пловдив

Knowledge level:



"Lili Ivanova sang in the amphitheater in Plovdiv"

6. Future work

Planned for future work are:

- Integration with other components of the CLaDA-BG project.
- PDF import, OCR, support for older and ancient languages.
- Export as docx and interactive documents for web page embedding.
- Spell checking, linguistic ambiguity by LLMs.
- For now, the system is tied to our requirements, but a modular approach can be implemented.
- Open-source version is considered after the production phase is achieved.

Bulgarian National Interdisciplinary Research e-Infrastructure for Resources and Technologies in favor of the Bulgarian Language and Cultural Heritage, part of the EU infrastructures CLARIN and DARIAH – CLaDA-BG