TEITOK API - Programmable DH Corpora

Janssen, Maarten
janssen@ufal.mff.cuni.cz
Charles University, Faculty of Mathematics and Physics, Czech Republic

Many (text) documents in DH projects contain more than just text: they contain added and deleted elements, changes of hands, links to facsimile images or sound data, footnotes, corrections, unclear passages, etc. This is why the TEITOK corpus environment (Janssen 2016) does not start from plain text, but rather from TEI/XML documents - the most well established format within DH, which offers solutions for all the above types of data. TEITOK is an open source platform that lets people collaboratively create, enhance, and annotate TEI/XML documents online, and end up with a searchable, integrated, annotated corpus. It has been used in a wide arrange of different types of corpora [1], in very diverse projects.

TEITOK was conceived to work on one document at a time, with integrated NLP solutions running on the server. However, in larger projects and infrastructures like LINDAT, more automatic interaction and flexibility is often required, which is why we developed an API. The TEITOK REST API lets you interact with your corpus remotely - to upload documents in various formats - automatically converted to TEI/XML; to render search results; to run an NLP pipeline on the server; or download the content of a corpus document, treat that content locally with NLP or manual annotation tools, and then upload the results back to the server, where the new or corrected annotations will be incorporated into the original TEI/XML document without destroying any of the potentially complex annotations already in the document. A description of the API is provided here: http://teitok.org/api.html

Since information can be both extracted and added, the TEITOK API turns TEITOK corpora into programmable corpora, not only in the sense used for instance in the Dracor corpus (Fischer, Frank, et al. 2019) that data can be extracted from the corpus in a programmable fashion, but also in a more extended sense in which the annotation of the corpus itself can be performed in a programmable fashion. How this works in practice is best explained by example.

Say we want to build a multimedia corpus of video interviews, where the initial transcription was done using subtitle tools, taken for example from linguisticteam.org [2], where they are stored in the subtitle format (SRT). The API allows you to upload this SRT file to the server, where it will be converted to TEI/XML [3]. And the TEITOK interface then provides an interface to view the data in a video-oriented manner, as shown in figure 1 [4].

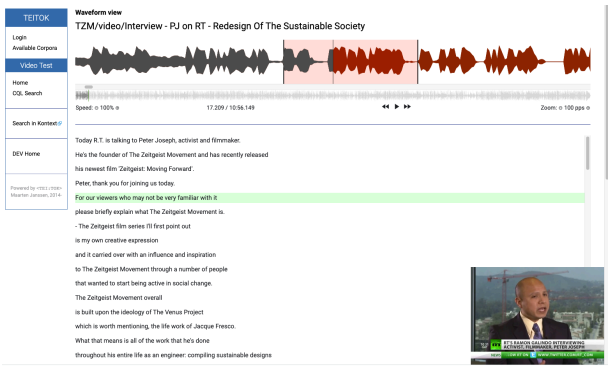

Figure 1. TEITOK interface for multimedia corpora

To enhance usability, we want to add additional annotations, like POS tags, dependency parses, named entities, etc. But not many NLP tools will be able to process TEI/XML data, let alone while preserving aligned video. The API solves this by letting you interact with the TEI in formats that are used in NLP workflows. You can for instance download the content in CoNLL-U, parse it using SpaCy [5], upload the data back, then download the data again in CAS [6], manually provide named entities using INCEpTION [7], and then upload again. Both uploads will incorporate the new information that was added locally into the existing TEI/XML file, while not breaking the video alignment. This way we end up with a TEI/XML file created from an SRT file, that is both time-aligned to the video and linguistically annotated.

A similar workflow can be used to start from a manuscript, use handwritten text recognition (HTR) tools like Transkribus [8] to get a corpus that is word-aligned to the images (which can be exploited in various ways in TEITOK), and then run text normalisation tools to normalise the orthography (added non-destructively), followed by running NLP tools over the normalised text. Or it can be used to automatically enhance metadata with local scripts, for instance to convert textual date descriptions in the teiHeader into an ISO date format, or to enrich place names with geolocation data. Or it can be used to create an automatic transcription using of a sound file using text-to-speech (TTS), linguistically annotate in TEITOK, and then download the result in EXMARALDA [9] or Praat [10] with tiers for the NLP annotations, to use elsewhere in sound-centred linguistic tools.

The TEITOK API thus makes it possible to combine different annotation tools on a single corpus, using the best tool and format for each separate aspect of the annotation. And then combine the result of these tools in a single TEI/XML based corpus that can be exploited in a number of different ways in TEITOK or elsewhere. Extracting only the relevant part of the content for each step makes for a more lightweight set-up than a traditional single-format workflow like WebLicht [11], where all the information has to be passed along in each step.

Notes

1. http://teitok.org/projects.html
2. https://files.linguisticteam.org/
3. The SRT could also be converted locally to TEI first if the TEITOK conversion is undesirable
4. https://quest.ms.mff.cuni.cz/teitok-dev/teitok/videotest/index.php?action=file&id=Peter_Joseph_on_RT.xml

5. https://spacy.io/
6. https://uima.apache.org/d/uimaj-current/apidocs/org/apache/uima/cas/CAS.html
7. https://inception-project.github.io/
8. https://readcoop.eu/transkribus/
9. https://exmaralda.org/en/
10. https://www.fon.hum.uva.nl/praat/
11. https://weblicht.sfs.uni-tuebingen.de/weblichtwiki/index.php/Main_Page

Bibliography

Fischer, Frank, et al. (2019): Programmable Corpora: Introducing DraCor, an Infrastructure for the Research on European Drama. In Proceedings of DH2019: "Complexities", Utrecht University.

Janssen, M. (2016): TEITOK: Text-Faithful Annotated Corpora. Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), Portorož, Slovenia.