

Analyzing TEI encoded texts with the TXM platform

Alexei Lavrentiev, Serge Heiden, Matthieu Decorde

▶ **To cite this version:**

Alexei Lavrentiev, Serge Heiden, Matthieu Decorde. Analyzing TEI encoded texts with the TXM platform. The Linked TEI: Text Encoding in the Web. TEI Conference and Members Meeting 2013, Oct 2013, Rome, Italy. halshs-01118120

HAL Id: halshs-01118120
https://shs.hal.science/halshs-01118120

Submitted on 18 Feb 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analyzing TEI encoded texts with the TXM platform

Lavrentiev, Alexei
Heiden, Serge
Decorde, Matthieu

TXM (http://sf.net/projects/txm) is an open-source software platform providing tools for qualitative and quantitative content analysis of text corpora. It implements the textometric (formerly lexicometric) methods developed in France since the 1980s, as well as generally used tools of corpus search and statistical text analysis (Heiden 2010).

TXM uses a TEI extension called "XML-TXM" as its native format for storing tokenized and annotated with NLP tools corpora source texts (http://sourceforge.net/apps/ mediawiki/txm/index.php?title=XML-TXM). The capacity to import and correctly analyze TEI encoded texts was one of the features requested in the original design of the platform.

However, the flexibility of the TEI framework (which is its force) and the variety of encoding practices make it virtually impossible to work out a universal strategy for building a properly structured corpus (i.e. compatible with the data model of the search and analysis engines) out of an arbitrary TEI encoded text or group of texts. It should nevertheless be possible to define a subset of TEI elements that would be correctly interpreted during the various stages of the corpus import process (for example, the TEI-lite tag set), to specify the minimum requirements to the document structure and to suggest a mechanism for customization. This work is being progressively carried out by the TXM development team, but it can hardly be successful without an input from the TEI community.

The goal of this paper is to present the way TXM currently deals with importing TEI encoded corpora and to discuss the ways to improve this process by interpreting TEI elements in terms of the TXM data model.

At present, TXM includes an "XML-TEI-BFM" import module developed for the texts of the Base de Français Médiéval (BFM) Old French corpus (http://txm.bfm-corpus.org) marked up according to the project specific TEI customization and guidelines (Guillot et al. 2010). With some adaptation, this module works correctly for a number of other TEI encoding schemas used by several projects: Perseus (http://www.perseus.tufts.edu/hopper), TextGrid (http://www.textgrid.de), PUC/Cléo (http://www.unicaen.fr/recherche/mrsh/document_numerique/outils), Frantext (http://www.frantext.fr), BVH (http://www.bvh.univ-tours.fr), etc. However, the use of tags that are not included in the BFM customization and the non respect of some particular constraints (such as a technique of tagging parts of words and of using strong punctuation within the editorial markup elements) may result in lower quality of the TXM corpus (e.g. errors in word counts, collocation analysis or inconvenient display of texts for reading) or even in a failure of the import process due to the limits of the tokenizer used in this module.

A more generic “XML/w+CSV” module allows importing any XML documents (not necessarily TEI) with the possibility to pre-annotate all or selected words using a <w> tag with an arbitrary set of attributes. This module is more robust in terms of producing a searchable corpus but it does not make any use of the semantics of TEI markup. For instance, no difference is made between the text and the header, the notes and variant encodings of the same text segment are all included in the text flow.

To improve the quality of the resulting corpus, it is necessary to “translate” the TEI markup into the various data categories relevant for the TXM data model. This model is relatively straightforward and relies to a large extent on that of the CWB CQP search engine (http://cwb.sourceforge.net). We have already presented the relevant data categories in some detail at the 2012 TEI Members Meeting (Heiden & Lavrentiev 2012) but this time we would like to adopt a more pragmatic approach related to the development of the TXM-TEI import modules.

A corpus is composed of a number of “text units” associated with a set of metadata used mainly to split the corpus in different ways and to perform contrastive analyses. A simple TEI file with one <text> element corresponds usually to a TXM text unit, and the useful metadata can be extracted from the <teiHeader> (or, alternatively, from a separate CSV table).

The second basic element of the TXM data model is the “lexical unit” (or the token), which may be a word or a punctuation mark carrying a number of properties (annotations) inherited from the source document (e.g. the language or a variant form) or generated during the import process (e.g. morphosyntactic description or a lemma suggested by an NLP tool). The properties of the lexical units can be easily searched and analyzed using the CQP search engine. TXM can import a corpus with pre-tagged lexical units but in most cases the tokenization is performed during the import process. In the latter case, it is necessary to pay special attention to the tags that may occur “inside” the tokens. These are typically line or page breaks, or some editorial markup (abbreviation marks, supplied letters, etc.). As far as the milestone-like empty elements are concerned, the TEI has recently adopted a general mechanism using the “break” attribute. As for the word-internal elements with textual content, it is recommended to pre-tag the words containing such elements using the <w> element before the import process.

The third element of the TXM data model is the intermediate structure of the text which can include sentences, paragraphs, divisions or any other continuously or sporadically marked up text segments. They are represented as XML elements, so proper nesting is required. They can be annotated by properties that can be used in a way similar to the text unit metadata. Intermediate structures can be used to separate “text planes” (such as titles vs. text body, direct speech of various characters in a drama, etc.). Although TXM is not designed for managing various readings in critical editions or stages of text evolution, the mechanism of text planes can be used to analyze and compare different text states or variants.

In the simplest case, a text can be represented as a chain of lexical units. This point of view is by all means relevant for word counts, collocation search and analysis, etc. If the source document contains editorial notes or variant encodings of the same text

segment (using <choice> or <app> mechanisms), it is necessary to treat them in one of the following ways:

- eliminate them completely from the search indexes;
- create a separate "text plane" for them and possibly relocate them to special text units or divisions;
- project variant readings as additional "properties" onto the lexical units of the main text chain.

The last but not the least aspect of the import process is building "editions" of corpus texts for convenient reading and displaying extended contexts of the search hits. This is where the rich TEI markup and the know-how of producing fancy-styled outputs may be particularly valuable. The objective is to make it possible to use a set of custom stylesheets (like those developed by Sebastian Ratz ones for the TEI consortium) to render these editions but this requires some further development to ensure compatibility with TXM's features of highlighting search hits and displaying properties of the lexical units. An intermediate solution is currently being experimented to allow the customization of the rendering of selected elements via the CSS class pointing mechanism.

The TXM team is interested in the feedback from any TEI projects willing to analyze their data with the TXM platform and is open to discussion on the improvement of the import modules and their documentation.

References:

Guillot, C., Heiden, S., Lavrentiev, A., Bertrand, L. (2010). Manuel d'encodage XML-TEI des textes de la Base de Français Médiéval, Lyon, Équipe BFM <http://bfm.ens-lyon.fr/article.php3?id_article=158>.

Heiden, S. (2010). "The TXM Platform: Building Open-Source Textual Analysis Software Compatible with the TEI Encoding Scheme." 24th Pacific Asia Conference on Language, Information and Computation. Éd. Kiyoshi Ishikawa Ryo Otoguro. Institute for Digital Enhancement of Cognitive Development, Waseda University, 2010. 389-398. <http://halshs.archives-ouvertes.fr/halshs-00549764>.

Heiden, S. & Lavrentiev, A. (2012). "Constructing Analytic Data Categories for Corpus Analysis from TEI encoded sources." TEI Conference 2012. College Station, TX, 7-10 November 2012. <http://idhmc.tamu.edu/teiconference/program/papers>.