On the Construction of a Personal Textbase with Document Markups and Metadata

Hsieh-Chang Tu
Department of Computer Science and Information Engineering, National Taiwan University

1. Introduction

Digital libraries are valuable resources for humanists to do research. They collect a great amount of digitalized texts and media, establish high-quality metadata that describe title, author, time and space, as well as many other properties for each repository item, and provide efficient search functions to help one find interesting items. In the past few years, however, researchers noticed insufficiency of these systems in several aspects. For instance, some may want to combine texts from different repositories, some require additional metadata not provided by the hosts, and some need to count and compare the frequencies of specific terms found in the texts. In short, researchers want to have more freedom to control the contents.

This paper addresses a solution to these problems by adopting a markup system and a platform for constructing personal textbases (text databases). A user first collects interesting texts from repositories, uses a markup tool to tag terms of concern, establishes text metadata, and then applies appropriate tools to build a textbase. In the following we use the term *document* to denote the unit of texts on concerns. We propose a data-integration process to combine all the documents, markups, and metadata into an all-in-one file. In particular, we discuss two online systems, MARKUS and DocuSky, that implement this process. The former is a system for tagging Chinese texts, and the latter is a platform that offers some other tools to accomplish the data integration task. DocuSky also allows one to upload the final integrated output to build a personal textbase for further applications.

2. The Proposed Data-Integration Process

Figure 1 shows the proposed process that takes text and metadata files as source inputs. It finally produces an XML file which combines document markups and metadata.

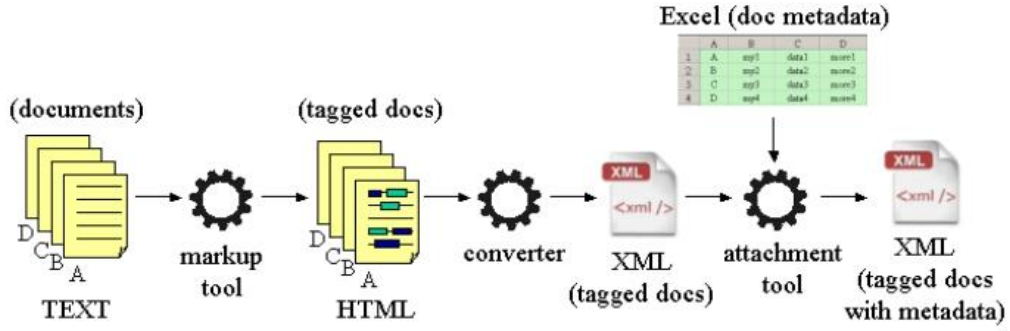

Figure 1. The proposed semi-pipelined process that combines document markups and metadata to form an integrated XML file ready for constructing a textbase. The user first employs a markup tool to tag source texts. The tagged outputs are taken as input of a converter to generate an XML file of tagged documents. An attachment tool then takes this XML file and a metadata file to yield the final XML output.

Since a term in texts can have different meanings, it is necessary to tag the terms of concern before counting and analyzing their occurrences. For a markup tool running under a modern browser, it is natural to store a tagged document in HTML. It is hard, however, to integrate a document in HTML format with its metadata since HTML is primarily intended for data presentation. Thus it is a good idea to convert the HTML markup into an XML file for later integration. On the other hand, people usually adopt a table structure, commonly stored in Excel format, to associate a document with its metadata. The first column of this table denotes document id, and the rest columns record metadata of this document. An attachment tool reads the converted XML and tries to "attach" metadata to the documents. This is done by repeating each row of the table, finding the target by the document id stored in the first column, and then updating its metadata by the values stored in the rest columns.

3. MARKUS

MARKUS is an online markup system that allows one to upload a text file in Chinese to tag terms the text. It offers convenient functions to help one tag terms manually or semi-automatically, and allows one to define term types in the tagging procedure.

With MARKUS, a tagged historical person name can have a CBDB1 id to identify this person uniquely, and a historical place name can have a TGAZ2 id or a PL id3 to distinguish it from others. Applications that require geo-information can use the place id to acquire its precise geo-location through the API provided by the database provider. MARKUS allows one to divide a document into many passages. Each passage can have a passage id as well as user comments. One can export a MARKUS file, at any time, and import it later to recover the markup process.

MARKUS regards each document as an independent file. There is little support for specifying the relations among documents. This can become a problem when one wants to analyze a large collection of documents.

1 China Biographical Database (https://projects.iq.harvard.edu/chinesecbdb/home)
2 TGAZ (Temporal Gazetteer) provides a normalized access to the Chinese Historical GIS placename database (http://maps.cga.harvard.edu/tgaz/).
3 PL id is the placename id provided by the DDBC Place Authority Database (http://authority.dila.edu.tw/place/).

4. DocuSky

DocuSky is a platform that allows a user to create one's own personal textbases. The design of DocuSky expects a textbase to support all the three retrieval functions provided by the THDL[4] system. These functions include fulltext search that allows one to find desirable documents with a set of keywords, post-query classification that shows the distribution of metadata values over a search result, and tag analysis that, with each tag type, returns a list of terms as well as their occurring frequencies.

In contrast to MARKUS that treats documents independently, DocuSky introduces a notion of corpus that groups a collection of documents. DocuSky regards each corpus as an independent entity, which means that all the search and analysis functions are applied within a corpus. However, to enable more flexibility for the future, DocuSky allows a textbase to have one or more corpuses.

DocuSky takes an XML file of proprietary format[5] as input to construct a textbase. It provides a markus converter and a metadata attachment tool to help one generate this XML. As an example in practice, one collects interesting texts first, tags the texts with MARKUS, and exports the result as HTML files. These files are then taken as inputs of the markus converter to yield an XML file with tagged documents. The markus converter supports an option to convert passages in a MARKUS file into documents. One can use this intermediate XML file to construct a textbase, albeit missing document metadata. If the user has established an Excel file with metadata, he/she can apply the metadata attachment tool to get an XML file that contains both document markups and metadata. This XML file can be used to build a DocuSky textbase which supports fulltext search, post-classification, and tag analysis.

---

4 Taiwan History Digital Library (http://thdl.ntu.edu.tw). In THDL and its related papers, *tag analysis* was often called *term analysis*. THDL supports three pre-defined tag types (PersonName, LocationName, and SpecificTerm). DocuSky supports both pre-defined and user-defined tag types.

5 Currently, DocuSky only supports XML with ThdlExportXml (THDL-exported XML) format.

Figure 2 shows a screenshot of the default retrieval tool in DocuSky. One may download the sample files6, which contains a tagged MARKUS file with 30 passages and an Excel metadata file, and apply the markus converter as well as the metadata attachment tool to re-construct this textbase.

Figure 2. A simple tag-analysis screenshot from a sample textbase. This textbase consists of 30 letters from YanWanli (楊萬里) with letter metadata. Each letter is marked as a passage with MARKUS, and each passage has been converted to a document by the converter. In this figure, the left panel lists the tag statistics of tag type Person. As shown in the list, a cbdb_id can refer to a person name and its alternatives occurred in documents. For instance, the cbdb_7197 refers to the person 周益國, which appears as either 周 or 益國 in the texts. The right panel displays the document metadata and text content.

5. Conclusion

This paper discusses the construction of a personal textbase. It proposes a process to combine document markups and metadata. In practice, we suggest to use MARKUS to tag documents, adopt Excel to record document metadata, and apply DocuSky tools to construct a textbase. A DocuSky textbase supports all the three retrieval functions (fulltext search, post-classification, and tag analysis) provided by conventional digital libraries. A noticeable thing is that now one gains a full control over contents in the textbase.

6 http://docusky.digital.ntu.edu.tw/docusky/data/markus-samples/MetadataAttachment-sample.zip