Corpus2Wiki: A MediaWiki based Annotation & Visualisation Tool for the Digital Humanities

Eleanor Rutherford[1], Wahed Hemati[2], Alexander Mehler[2]

Trinity College Dublin[1]
Text Technology Group, Goethe University Frankfurt am Main[2]

Abstract

In this paper, we present Corpus2Wiki, a tool which automatically creates a MediaWiki site for a given corpus of texts. The texts, along with automatically generated annotations and visualisations associated with them, are displayed on this MediaWiki site, locally hosted on the user's own machine. Several different software components are used to this end - Docker for ease and consistency of deployment, MediaWiki for the core engine, TextImager Client for the generation of annotations and a number of existing, and as well as extended, MediaWiki extensions for the visualisations. This tool was specifically designed for use within the interdisciplinary field of the Digital Humanities, as it provides a visual analysis and representation of texts via a tool which require no programming or advanced computational knowledge and uses an interface already well-known within the Digital Humanities Community, namely MediaWiki.

1 Introduction

In order to demonstrate the applications of our tool in the Digital Humanities, first we need to define what is meant by this term. As Svensson (2016) observes, "the field of digital humanities has a reputation for being difficult to define and for being preoccupied with defining itself." Despite this, Burdick et. al (2012, p.122) provide a concise, eloquent definition: "Digital Humanities refers to new modes of scholarship and institutional units for collaborative, transdisciplinary, and computationally engaged research, teaching, and publication."

Furthermore, they discuss the ways in which the conception and design of projects in the Digital Humanities can, and indeed should, be influenced by said "computational engagement," focusing on four methods; namely, curation, analysis, editing and modelling. Our project combines all four of these methodologies to provide a useful and applicable tool in several areas of this interdisciplinary field: it organizes and (locally and permanently, that is, not deleted at the end of a web session) displays texts of a given corpus using a well-known, widely-used interface and framework, MediaWiki. (*Curation*) The texts are *analyzed* so as to output grammatical information (i.e. POS tags and lemmata) above each word in the

Veröffentlicht durch die Gesellschaft für Informatik e.V. 2018 in
M. Burghardt & C. Müller-Birn (Hrsg.): INF-DH 2018 – Workshopband, 25. Sept. 2018, Berlin
 https://doi.org/10.18420/infdh2018-08

form of a tool tip, as well as statistical information pertaining to the texts (i.e. POS frequency and type), displayed as visualisations "in order to give graphical legibility to analytical results." (Burdick et. al, 2012)

Once the texts are displayed as MediaWiki pages, they can be *edited* using a particular syntax, and the text display (and visualisations) will update in real time. Finally, the last methodology, *Modelling*, ties in with the first and relates to how the data is structured and presented, that is to say, in a way that makes sense for the particular type of data and the function of the tool - in our case, the decision to use the MediaWiki interface to implement this tool. Using the MediaWiki interface allows an attractive, easy to edit display of texts, in a format already familiar to many in the Digital Humanities. The use of the TextImager Client also means that the tool can potentially access the hundreds of existing text analysis tools already provided by the client (and the visualisations associated with them), allowing relatively easy expansion of our tool to include a wide range of other textual analyses and graphics e.g. use of the tool by historians, using the tool to plot a historical figure's life (given a biographical text for example), both geographically and chronologically, by teachers, using the tool didactically in a language instruction context (evaluating and/or analyzing student texts for example) and by linguists, using the tool purely as a linguistic analysis service, see Future Work for more.

Leaving aside the scope for extension, how can this tool be used in its current capacity and with its current functionality? (For each text; provision of lemmata, POS tags and a frequency histogram for the tags.) A potential research question which could be answered by the tool is as follows: A linguist is interested in grammatical differences in language used in different domains for example scientific fields vs. fictional texts. They want to find out whether the scientific texts in their corpus contain more verbs (for example) than in the fictional texts. Alternatively, they could have a corpus containing texts from different time periods and wish to investigate a similar hypothesis: are the texts from earlier time periods more noun heavy, for instance, than those written in recent years? Our tool would be suitable for such a task for several reasons: All the texts are analysed and stored *locally* on the MediaWiki site thus enabling the linguist to navigate *between* texts, rather than analysing the texts one at a time, as is the case with many other annotation tools, see Related Work below for more. The frequency histograms for the POS tags in each text provides the information required for the linguist in a visually friendly manner and allows the linguist to acquire the information they need at a glance.

2 Related Work

Several visualisation-based annotation tools (which provide similar functionality as Corpus2Wiki) already exist, for example WebAnno[1], TreeAnnotator (Helfrich et.al, 2018), FLAT[2] (FoLiA Linguistic Annotation Tool), WebNLP (Burghardt et. al, 2014) and

[1] https://webanno.github.io/webanno/

[2] https://github.com/proycon/flat

Wikidition (Mehler et.al, 2016). WebAnno is perhaps the closest to our tool in terms of ease of installation and use, providing both executable JAR installation and install via Docker options. However, as it provides a framework with which users can manually create and manage annotations, rather than generating them automatically (as Corpus2Wiki does), the two are not directly comparable. Similarly, the task of TreeAnnotator is fundamentally different to that of Corpus2Wiki, in that the type of annotation it deals with falls within the domain of discourse annotation, as opposed to the grammatical annotation Corpus2Wiki produces. In comparison, Wikidition provides an in-depth textual analysis on syntagmatic and paradigmatic level between texts, sentences and word, also displayed in a Wiki format. Although a very valuable and useful tool, it provides a level of linguistic information that is perhaps excessive for the average user in the Digital Humanities. Corpus2Wiki then intends to provide text analysis with visual support in this Wiki format while remaining accessible to users across all domains in this field, not just linguists.

FLAT is an annotation tool that is more developed than ours in terms of the range of features offered to the user, for example, different modes of editor are provided, more grammatical data is available and a GUI Annotation Editor is included. However, the install of FLAT requires quite advanced technical skills and knowledge in order to carry out the multiple installation and configuration processes via command line, effectively reducing the number of potential users. We avoid this fate by deploying our tool via Docker, thus increasing accessibility of our tool to users with only very basic computer literacy and thereby hopefully reaching users unable to use tools such as FLAT due to this lack of technical knowledge. FLAT also requires input texts to be of the specific form FoLiA, further narrowing its usage by a larger demographic. In comparison, Corpus2Wiki accepts simple text files (and then converts them to a single xml dump), further demonstrating the accessibility of the tool.

WebNLP is the annotation tool probably closest to ours with regards to the service it provides. How then do the two tools differ, and what makes Corpus2Wiki necessary and valuable among the host of different annotation tools which already exist? WebNLP is a web service which draws upon Python NLTK[3] for the annotation and Voyant Tools[4] (in itself a popular text analysis and visualisation tool in the Digital Humanities) for visualisations (although at the time of writing, this visualisation functionality is not available). That WebNLP is a web service (which is not open source) means the tool is hosted online and no installation is required. However, the *local* installation of Corpus2Wiki makes it ideal for the analysis of texts which users do not wish to publish online because of licence restrictions etc. Its open source nature (in the spirit of the MediaWiki ethos) also means the source code can be edited, extended or reshaped by anybody to fit their own purpose, another reason why we decided to implement using MediaWiki (and publish our entire tool on GitHub.)

Another fundamental difference between the two tools is the ability of Corpus2Wiki to analyze more than one text at a time i.e. a corpus, functionality not included in WebNLP or comparable tools. This allows our tool to create “links” or establish relationships between

[3] https://www.nltk.org/

[4] https://voyant-tools.org/

pages. An example of an application of this functionality is the creation of Category Pages within the MediaWiki site, allowing users to view texts in their corpus grouped according to subject (thematised by the Dewey Decimal Classification), a clear advantage of our MediaWiki implementation. Furthermore, the MediaWiki interface, already familiar to many in the Digital Humanities, provides an attractive, relatively easy to navigate front end, whereby users can easily switch between texts by using the links to the individual texts on the Corpus page. Our dynamic visualisations (in the current release, frequency histograms, POS tag highlighting and lemmata as tool tips - to be extended;) further enhance this attractive front-end design and provide a clear, accessible and useful graphical representation of often complex textual information.

The implementation of MediaWiki also creates the possibility of easily extending and improving the tool by way of utilising custom, as well as existing, MediaWiki extensions, (see Future Work) increasing the potential reach and scope of the tool by users (and indeed by developers wishing to adapt the tool for custom usage) from a wide variety of fields, and with varying needs and uses for the tool.

3 Implementation

The implementation is as follows:

The MediaWiki site is installed (locally) using Docker, specifically using docker-compose via yml file. Some input must be manually input by the user, but as the MediaWiki installation wizard can be used to this end and as we have provided complete documentation detailing the process, it should not prove to be technically challenging. In future releases of the tool, it is intended to fully automate this process. The use of Docker means that the install will be functional across all operating systems and will not require the manual installation of necessary dependencies e.g. PHP, MariaDB (or a similar database) and Apache (or comparable server) (see Figure 1). It also eliminates the risk that the install will not work because the user has a different version of some software on their machine. This significantly reduces the expertise, computational knowledge and time necessary to install MediaWiki, and thus our tool.

After the MediaWiki install, the tool is run - the corpus (to be found in the home directory of the user's machine) is automatically preprocessed (e.g. lemmatized and POS-tagged) by the TextImager Client (Hemati et. al, 2016) which produces an XML file in a format processable by MediaWiki. The TextImager Client JAR is included in the tool image, thereby removing the need for the user to have TextImager installed locally on their machine.

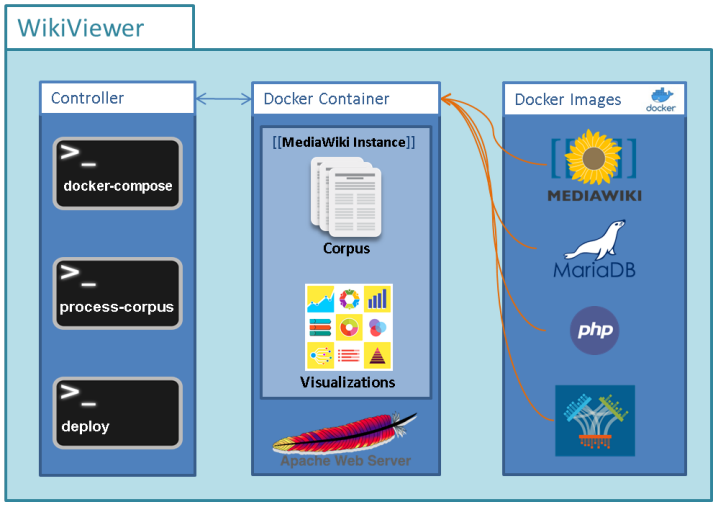

Figure 1. This Figure shows the architecture of WikiViewer. To provide a system independent installation procedure, Docker Images are utilized to build a Docker container containing the necessary components for deployment of WikiViewer. The container holds the processed Corpus along with the Visualizations.

Using behind the scenes MediaWiki functionality (namely the script "importDump.php"), the XML file produced by the TextImager Client is imported into the MediaWiki site as individual pages - each text is displayed as a separate page and the corpus is also displayed as a single page, containing links to each unique page corresponding to its particular text.

Visualisations relating to the text (created using MediaWiki extensions, for example SimpleToolTip[5] and Graph[6] in our current release) are shown on each text page. Currently our tool supports histogram frequency visualisations and token highlighting capabilities, with annotations generated by a Language Segmenter, Lemmatizer and POS Tagger. However, as the tool generates annotations and textual information from the TextImager Client, any of the hundreds of tools (including sentiment analysis tools, text similarity tools, parsers, named entity recognisers and more) in any of the languages provided by the TextImager Client could potentially be available for use within the tool. Visualisations pertaining to this textual information can be loaded directly from the TextImager Client and new visualisations can also be developed as new or existing MediaWiki extensions, one of the many advantages of implementing this tool via MediaWiki and the TextImager Client, see Future Work for more. Each text can also be easily edited using MediaWiki edit functionality coupled with our easy to use syntax for example, the following syntax would be used to display the token "annotations" {{#tip-text: annotations |lemma:annotation,pos:NN}}.

[5]https://www.mediawiki.org/wiki/Extension:SimpleTooltip
[6] https://www.mediawiki.org/wiki/Extension:Graph

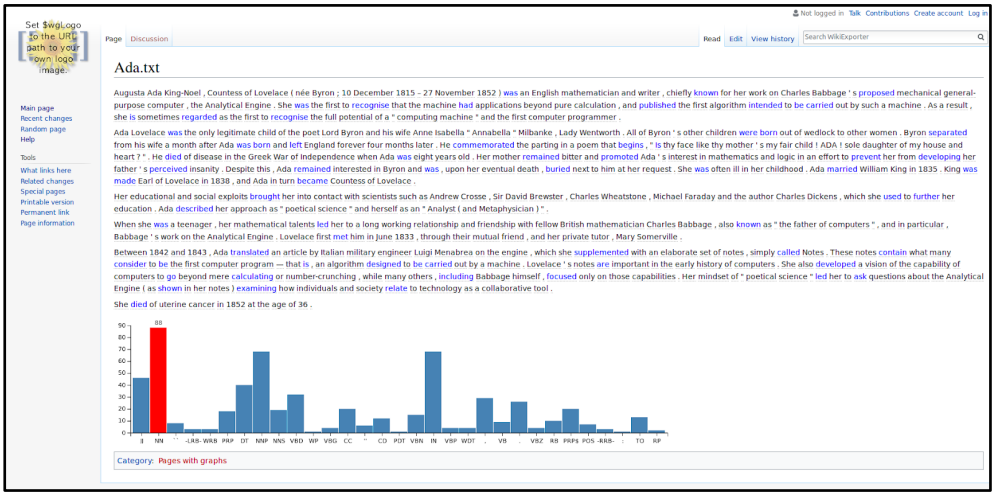

Figure 2: Example text based on Wikipedia biography of Ada Lovelace.

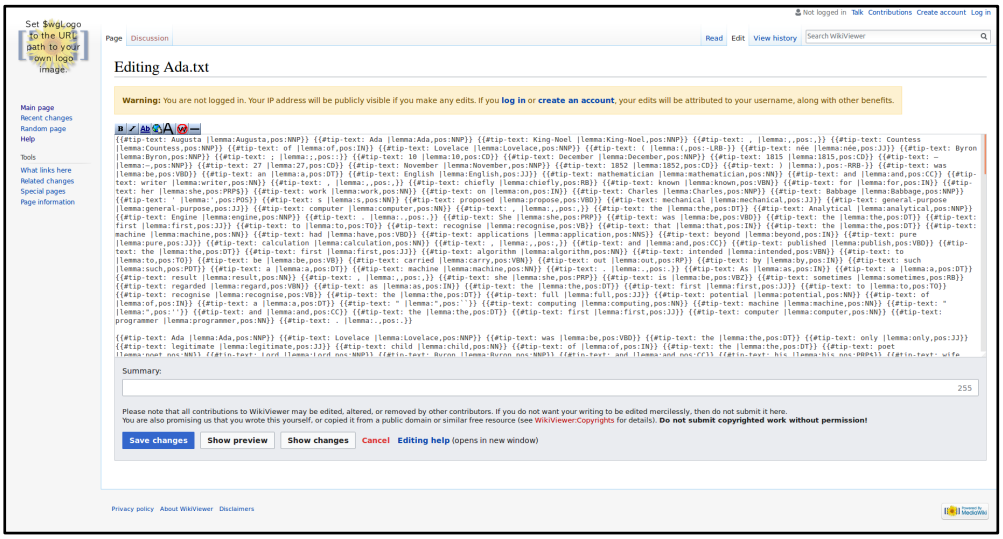

Figure 3: Edit page of text using the MediaWiki edit functionality. Each edition is versionised in the MediaWiki history, making it possible to review and if needed to undo edition. MediaWiki is a multi user system, so multiple users can edit the same page.

All of this functionality, along with the creation of the MediaWiki pages corresponding to the texts in the corpus, is generated automatically when the tool is executed - all necessary steps for functionality of the tool are carried out (automatically) via command line when the tool is executed.

4 Future Work

Our implementation of Corpus2Wiki via MediaWiki and the TextImager Client creates boundless opportunity for the expansion and development of this tool. We envision three main points of development for our tool in the near future: firstly, the number of tools and textual information, and visualisations to illustrate them should be increased. Secondly, the tool should be further automated to create the ultimate user-friendly tool. Thirdly, the number of languages supported should be increased to facilitate international and interlingual use of the tool. In order to achieve our first goal, we aim to provide a wider range of automatically generated textual information from the TextImager Client, represented primarily by visualisations and graphics.

Visualisations can be imported from the TextImager Client itself, but can also be implemented as MediaWiki extensions, providing further possibilities for expansion and improvement.

Figure 4: EasyTimeline visualisation, https://www.mediawiki.org/wiki/Extension:EasyTimeline

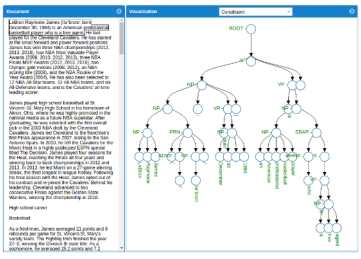

Figure 5: Constiuent Parse Tree, TextImager 2016

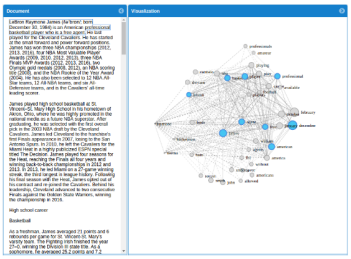

Figure 6: Semantic Relation Graph, TextImager 2016

For example, the Maps Extension could be implemented in order to display geographical information mentioned in a text, the IssueTracker Extension could be used to add project management functionality to the tool, or the chronological development of a text could be tracked and plotted using a timeline extension such as EasyTimeline or ChapTimeline. We also hope to add word embedding functionality as a custom developed MediaWiki extension.

Secondly, we would like to further automate the process. As it stands, a potential user must still input some data into the MediaWiki wizard to complete the installation.

Although this should pose no problems to users even with very basic computer knowledge, as we have fully documented the process, it would be our hope to fully automate this process in the future, thereby allowing users to simply click a single button to run the tool.

Lastly, support for other languages and file types should also be introduced (currently English and text files are the only available options). Again, this goal is very much achievable using existing TextImager capabilities, namely, its support for 10 different languages and numerous file types. With all these extensions and developments, we envision a far-reaching linguistic analysis tool, to be used in the Digital Humanities for many years to come.

5 System Demonstration

The source code is open source and can be found at the GitHub repository:
https://github.com/texttechnologylab/textimager-wikidition

The Repository contains the code for setting up a Docker instance of Corpus2Wiki. A running example can be found at:

https://textimager.hucompute.org/corpus2wiki/

6 References

Burdick, A., Drucker, J., Lunenfeld, P., Presner, T., & Schnapp, J. (2016). *Digital_Humanities.* Retrieved July 02, 2018, from https://mitpress.mit.edu/books/digitalhumanities

Burghardt, M., Pörsch, J., Tirlea, B., & Wolff, C. (2014, October 8–10) *WebNLP: An Integrated Web-Interface for Python NLTK and Voyant.* In: Ruppenhofer, Josef and Faaß, Gertrud, (eds.) Proceedings of the 12th edition of the KONVENS conference, Hildesheim, Germany.

Helfrich, P., Rieb, E., Abrami, G., Lücking, A., & Mehler, A. (2018, May 7–12) *"TreeAnnotator: Versatile Visual Annotation of Hierarchical Text Relations,"* in Proceedings of the 11th edition of the Language Resources and Evaluation Conference, Miyazaki, Japan.

Hemati, W., Uslu, T., & Mehler A., (2016, December 11–16) *"TextImager: a Distributed UIMA-based System for NLP,"* in Proceedings of the COLING 2016 System Demonstrations, Osaka, Japan.

Mehler, A., Gleim, R., vor der Brück, T., Hemati, W., Uslu, T., & Eger S., "Wikidition: Automatic Lexiconization and Linkification of Text Corpora," *Information Technology*, pp. 70-79, 2016.

Mehler, A., Wagner, B., and Gleim R., *Wikidition: Towards A Multi-layer Network Model of Intertextuality* in Proceedings of DH 2016, 12-16 July, 2016.

Svensson, P. (2016). *Big Digital Humanities: Imagining a meeting place for the humanities and the digital*. Ann Arbor: University of Michigan Press.