PICCL: Philosophical Integrator of Computational and Corpus Libraries

Martin Reynaert[1 2] **Maarten van Gompel**[2] **Ko van der Sloot**[2] **Antal van den Bosch**[2]
TiCC / Tilburg University[1] CLST / Radboud University Nijmegen[2]
The Netherlands
mreynaert|M.vanGompel|K.vanderSloot|a.vandenbosch@let.ru.nl

Abstract

CLARIN activities in the Netherlands in 2015 are in transition between the first national project CLARIN-NL and its successor CLARIAH. In this paper we give an overview of important infrastructure developments which have taken place throughout the first and which are taken to a further level in the second. We show how relatively small accomplishments in particular projects enable larger steps in further ones and how the synergy of these projects helps the national infrastructure to outgrow mere demonstrators and to move towards mature production systems. The paper centers around a new corpus building tool called PICCL. This integrated pipeline offers a comprehensive range of conversion facilities for legacy electronic text formats, Optical Character Recognition for text images, automatic text correction and normalization, linguistic annotation, and preparation for corpus exploration and exploitation environments. We give a concise overview of PICCL's components, integrated now or to be incorporated in the foreseeable future.

1 Introduction

The transition from CLARIN-NL (Odijk, 2010) to its successor CLARIAH[1] offers an opportunity to assess past and future CLARIN activities in the Netherlands. We give an overview of text infrastructure developments which have taken place throughout the first and are set to be taken to a further level in the second. The demonstrator built in CLARIN-NL Call 4 project @PhilosTEI is now in CLARIAH to grow into the full-fledged production system PICCL. This acronym stands for 'Philosophical Integrator of Computational and Corpus Libraries', the first term of which signifies 'well-considered' and is hoped for its users to grow to mean 'practical'.

2 PICCL: an overview

PICCL constitutes a complete workflow for corpus building. It is to be the integrated result of developments in the CLARIN-NL project @PhilosTEI, which ended November 2014, further work in NWO 'Groot' project Nederlab[2], which continues up to 2018, and in CLARIAH, which runs until 2019.

2.1 What went before

At Tilburg University, the Netherlands, work was started on building web applications and services for the CLARIN infrastructure in 2011 in project CLARIN-NL TICCLops (Reynaert, 2014b). In this CLARIN-NL Call 1 project the idea to provide text normalization and spelling/OCR post-correction facilities as an 'online processing service' – hence the -ops in the project name – spawned the idea of building a generic system for turning linguistic command-line applications into RESTful web services and web applications. This begat CLAM, the Computational Linguistics Application Mediator (van Gompel and Reynaert, 2014), which TICCLops builds upon. CLAM[3] has been adopted widely within the CLARIN-NL community and underlays the Dutch-Flemish cooperation in the CLARIN-NL infrastructure project

[1] http://www.clariah.nl/en/

[2] https://www.nederlab.nl/onderzoeksportaal/

[3] https://proycon.github.io/clam

TTNWW, in which available tools for both text as well as speech are turned into web services and subsequently united in a workflow management system (Kemps-Snijders et al., 2012).

The storage and exchange of linguistically annotated resources requires a modern and expressive format capable of encoding a wide variety of linguistic annotations. A solution has been devised in the form of FoLiA, short for “Format for Linguistic Annotation” (van Gompel and Reynaert, 2013). FoLiA provides a generic single-solution XML format for a wide variety of linguistic annotations, including lemmata, part-of-speech tags, named-entity labels, shallow and deep syntactic structure, spelling and OCR variation, etc. Furthermore, it provides an ever-expanding software infrastructure to work with the format. The format was adopted by the large corpus building effort for Dutch, the SoNaR project (Oostdijk et al., 2013) in the STEVIN programme, as well as other other projects. In order to provide uniform linguistic annotations for this 540 million word token reference corpus of contemporary, written Dutch, Frog[4] (Van den Bosch et al., 2007), a suite of various natural language processing tools for Dutch based on the TIMBL classifier (Daelemans et al.,), was further developed.

2.2 PICCL: system overview

PICCL aims to provide its users with the means to convert their textual research data into an easily accessible, researchable corpus in a format fit for the future. A schematic overview of the PICCL pipeline and some of the planned extensions is shown in Figure 2.2.

Input can be either images or text. Images may be e.g. the scanned pages of a book in DjVu, PDF or TIFF formats. Text images are converted into electronic text by Tesseract[5]. Text may be plain, in various word-processing formats, embedded in PDF, or in OCR engine output formats; i.e. hOCR HTML, Page XML, or Alto XML. Alto XML is the major text format in the large digital text collections aggregated by the Dutch National Library (KB). The conversion tool FoLiA-alto developed for the Nederlab project allows for direct harvesting from the KB. To overcome the severe acceptable input format limitations of (Jongejan, 2013), PICCL is to be equipped with convertors for a panoply of document formats. We intend to incorporate OpenConvert[6], another CLARIN-NL web service. FoLiA XML is PICCL’s pivot format. The workflow can handle texts in a broad range of –currently– European languages. Provisions are available for dealing with old print or diachronical language variation.

Output text is in FoLiA XML[7]. The pipeline will therefore offer the various software tools that support FoLiA. Language categorization may be performed by the tool FoLiA-langcat at the paragraph level. TICCL or ‘Text-Induced Corpus Clean-up’ performs automatic post-correction of the OCRed text. Dutch texts may optionally be annotated automatically by Frog, i.e. tokenized, lemmatized and classified for parts of speech, named entities and dependency relations. The FoLiA Linguistic Annotation Tool (FLAT)[8] will provide for manual annotation of e.g. metadata elements within the text – for later extraction. FoLiA-stats delivers n-gram frequency lists for the texts’ word forms, lemmata, and parts of speech. Colibri Core[9] allows for more efficient pattern extraction, on text only, and furthermore can index the text, allowing comparisons to be made between patterns in different (sub)corpora. BlackLab[10] and front-end WhiteLab[11], developed in the OpenSoNaR project[12] (Reynaert et al., 2014), allow for corpus indexing and querying. Convertors to other formats, e.g. TEI XML, for allowing scholars to build critical editions of books, will be at hand.

PICCL is to be available to all researchers in the CLARIN infrastructure and is hosted by certified CLARIN Centre INL in Leiden. PICCL is to have a highly intuitive user-friendly interface in order to allow even the most computer-weary user to obtain texts in a corpus-ready, annotated format. Its predecessor, the @PhilosTEI system, provides two distinct interfaces: the more generic interface type

[4] http://ilk.uvt.nl/frog

[5] https://github.com/tesseract-ocr

[6] https://github.com/INL/OpenConvert

[7] https://proycon.github.io/folia

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

[9] https://proycon.github.io/colibri-core

[10] https://github.com/INL/BlackLab/wiki

[11] https://github.com/TiCCSoftware/WhiteLab

[12] http://opensonar.clarin.inl.nl

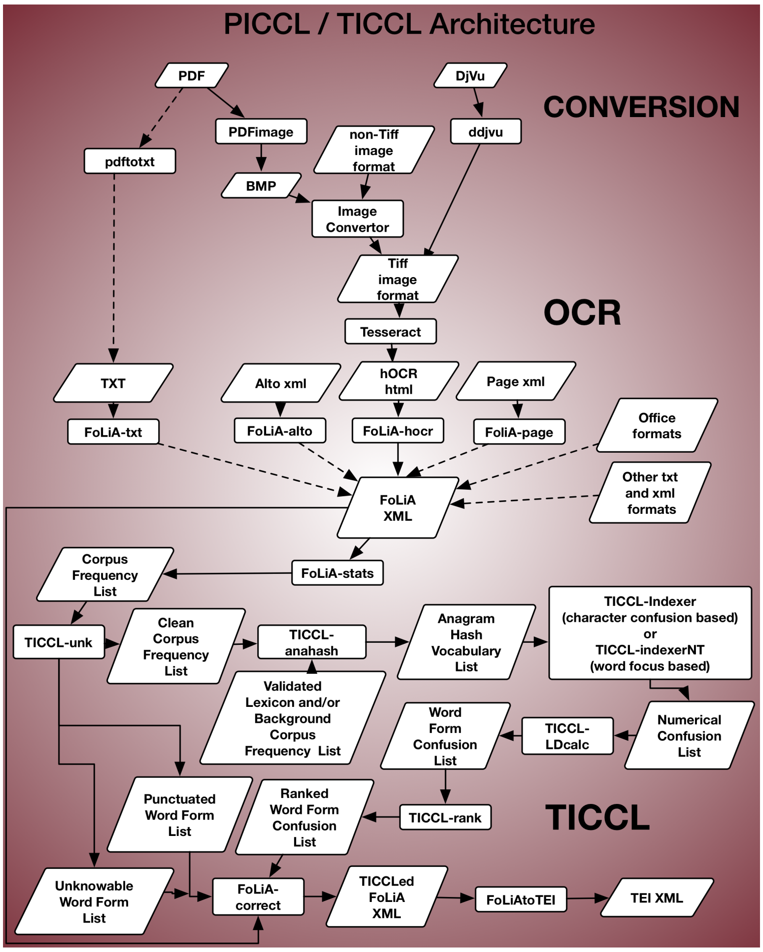

Figure 1: A schematic overview of the PICCL pipeline, dotted lines signify future extensions

that comes with CLAM [13] as well as a more end-user-oriented web interface that was custom made for the @PhilosTEI project, according to the specifications of the end users, i.e. philosophers[14].

PICCL continues to be developed with the users firmly in mind. We aim to make the user-friendly system available as a large black box that processes a book's images into a digital version with next to no user intervention or prior knowledge required. At the same time we also want to equip PICCL with the necessary interface options to allow more sophisticated users to address any sub-module or combination of sub-modules individually at will.

Future developments in CLARIAH are that Frog is to be made more easily retrainable, e.g. for older varieties of Dutch. It is also to be trained for both present-day English and German.

2.3 PICCL in comparison

In contrast to the CLARIN-NL TTNWW workflow (Kemps-Snijders et al., 2012) in the Taverna[15] framework, PICCL is implemented as a single and efficient pipeline, rather than a collection of many interconnected webservices. PICCL's focus is on the end-user who has an interest in the pipeline as a whole rather than its individual parts. This approach avoids network overhead, which can be a significant bottleneck in dealing with large corpus data. It still allows for distributional use of the available hardware through load-balancing, and still allows for the whole to be available as a RESTful webservice, through CLAM, for

[13] `http://ticclops.clarin.inl.nl`

[14] `http://philostei.clarin.inl.nl`

[15] `http://www.taverna.org.uk`

automated connectivity. Another major difference between TTNWW and PICCL is that the latter allows for better control over and handling of the text document flow. A TTNWW workflow offers sequential handling of documents by the various web services only, i.e. every single input file is processed in turn by each service and passed on to the next. In contrast, the PICCL wrapper allows for flexible handling of numbers of input/output files, taking e.g. x PDF input files apart into y (where $y \geq x$) image files to be sent to the OCR engine Tesseract, then presenting the y OCRed files as a single batch to TICCL which eventually corrects the y FoLiA XML files to be collated into a single output FoLiA XML and also, if the user so desires, a TEI XML output e-book.

Another solution for NLP workflows is provided by the Weblicht project (Hinrichs et al., 2010), developed in the scope of CLARIN-D. Both Taverna and Weblicht are generic workflow frameworks and are more suited for a larger number of interconnected webservices. PICCL is a single, more monolithic, workflow, albeit heavily parametrised. Weblicht also centers around their own TCF file format whilst our solutions are deliberately FoLiA-based because it can better express spelling correction and lexical normalization.

3 TICCL: an overview

A major component of the PICCL pipeline is Text-Induced Corpus Clean-up or TICCL, a system for unsupervised spelling correction and lexical normalisation or post-correction of OCRed corpora. TICCL is now multilingual and diachronic. In contrast to e.g. Vobl et al. (2014), TICCL aims at fully automatic post-correction. It was shown to outperform VARD2 (Baron and Rayson, 2008) in Reynaert et al. (2012) in the task of spelling normalization of historical Portuguese.

3.1 TICCL: current implementation

TICCL currently consists of a wrapper (written in Perl) around efficient multithreaded modules (in C++). Two of these modules, respectively the first and last of the TICCL pipeline, work on and require FoLiA XML input. The intermediate modules are TICCL-specific, and do not work on running text but rather on lists containing either words and frequency information or anagram hash values derived from corpus and lexicon words. The main publication on how TICCL operates is (Reynaert, 2010). Reynaert (2014a) offers more details on its current implementation and an in-depth evaluation on historical Dutch. This shows that when equipped with the most comprehensive historical lexicons and name lists, as well as with the word frequency information derived from a large background corpus of contemporary books, TICCL achieves high precision and useful recall. After fully automated correction, the word accuracy of the gold standard book experimentally corrected was raised from about 75% to 95%.

3.2 TICCL and language-specific lexical resources

TICCL relies on word and n-gram frequencies derived from the corpus to be cleaned. It can also be provided with further word form frequencies derived from e.g. another – possibly very large – background corpus. For languages other than Dutch the system is currently equipped with open source lexicons only. More importantly, PICCL will allow its users to equip the system with their own lexical resources of choice through the simple expedient of uploading them.

4 Conclusion

We have given an overview of work delivered and ongoing on PICCL, a comprehensive corpus building work flow. The system is geared to be equipped with the best available solutions for the sub-problems it is meant to solve. It is highly user-friendly, shielding the user to the highest extent from the intricacies of the many software modules it is composed of, asking only for the most minimal user input possible. The Nederlab project as prime user of the system is set to ‘piccl’ a great many diachronic corpora of Dutch. We hope PICCL will enable anyone to build their own personal text corpora and to derive the utmost benefit from them.

Acknowledgements

The authors acknowledge having been funded by several projects in CLARIN-NL and to be funded by the Dutch national CLARIN project CLARIAH. Martin Reynaert further acknowledges being funded by NWO in the project Nederlab.

References

Alistair Baron and Paul Rayson. 2008. VARD2: A tool for dealing with spelling variation in historical corpora. In *Proceedings of the Postgraduate Conference in Corpus Linguistics*.

Walter Daelemans, Jakub Zavrel, Ko Van der Sloot, and Antal Van den Bosch. TiMBL: Tilburg Memory Based Learner, version 6.3, Reference Guide, year = 2010. Technical Report ILK 10-01, ILK Research Group, Tilburg University.

Marie Hinrichs, Thomas Zastrow, and Erhard W. Hinrichs. 2010. Weblicht: Web-based LRT Services in a Distributed eScience Infrastructure. In Nicoletta et al. Calzolari, editor, *LREC*. European Language Resources Association.

Bart Jongejan. 2013. Workflow Management in CLARIN-DK. In *Proceedings of the workshop on Nordic language research infrastructure at NODALIDA 2013*, volume 089 of *NEALT*, pages 11–20.

Marc Kemps-Snijders, Matthijs Brouwer, Jan Pieter Kunst, and Tom Visser. 2012. Dynamic web service deployment in a cloud environment. In Nicoletta Calzolari et al., editor, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 2941–2944, Istanbul, Turkey. ELRA.

Jan Odijk. 2010. The CLARIN-NL project. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation, LREC-2010*, pages 48–53, Valletta, Malta.

Nelleke Oostdijk, Martin Reynaert, Véronique Hoste, and Ineke Schuurman. 2013. The construction of a 500-million-word reference corpus of contemporary written Dutch. In *Essential Speech and Language Technology for Dutch: Results by the STEVIN-programme*, chapter 13. Springer Verlag.

Martin Reynaert, Iris Hendrickx, and Rita Marquilhas. 2012. Historical spelling normalization. A comparison of two statistical methods: TICCL and VARD2. In Francesco Mambrini, Marco Passarotti, and Caroline Sporleder, editors, *Proceedings of ACRH-2*, pages 87–98. Lisbon: Colibri.

Martin Reynaert, Matje van de Camp, and Menno van Zaanen. 2014. OpenSoNaR: user-driven development of the SoNaR corpus interfaces. In *Proceedings of COLING 2014: System Demonstrations*, pages 124–128, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Martin Reynaert. 2010. Character confusion versus focus word-based correction of spelling and OCR variants in corpora. *International Journal on Document Analysis and Recognition*, 14:173–187.

Martin Reynaert. 2014a. Synergy of Nederlab and @PhilosTEI: diachronic and multilingual Text-Induced Corpus Clean-up. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. ELRA.

Martin Reynaert. 2014b. TICCLops: Text-Induced Corpus Clean-up as online processing system. In *Proceedings of COLING 2014: System Demonstrations*, pages 52–56, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Antal Van den Bosch, Gertjan Busser, Sander Canisius, and Walter Daelemans. 2007. An efficient memory-based morpho-syntactic tagger and parser for Dutch. In P. Dirix et al., editor, *Computational Linguistics in the Netherlands: Selected Papers from the Seventeenth CLIN Meeting*, pages 99–114, Leuven, Belgium.

Maarten van Gompel and Martin Reynaert. 2013. FoLiA: A practical XML Format for Linguistic Annotation - a descriptive and comparative study. *Computational Linguistics in the Netherlands Journal*, 3.

Maarten van Gompel and Martin Reynaert. 2014. CLAM: Quickly deploy NLP command-line tools on the web. In *Proceedings of COLING 2014: System Demonstrations*, pages 71–75, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Thorsten Vobl, Annette Gotscharek, Ulrich Reffle, Christoph Ringlstetter, and Klaus Schulz. 2014. PoCoTo - An Open Source System for Efficient Interactive Postcorrection of OCRed Historical Texts. In *Proceedings of Datech 2014*. ACM.