Controlled Natural Language for Semantic Annotation

Brian Davis and Pradeep Varma and Siegfried Handschuh and Laura Dragan[1] and Hamish Cunningham[2]

[1] Digital Enterprise Research Institute, National University of Ireland, Galway
{brian.davis, pradeep.varma, siegfried.handschuh, laura.dragan }@deri.org
[2] Sheffield NLP Group, University of Sheffield
hamish@dcs.shef.ac.uk
Extended Abstract

1 Introduction

Richly interlinked, machine-understandable data constitute the basis for the Semantic Web and by extension the Social Semantic Desktop[2]. Manual semantic annotation is a complex and arduous task both time-consuming and costly often requiring specialist annotators. (Semi)-automatic annotation tools attempt to ease this process by detecting instances of classes within text and relationships between classes, however their usage often requires knowledge of Natural Language Processing(NLP) and/or formal ontological descriptions. This challenges researchers to develop user-friendly annotation environments within the knowledge acquisition process. Controlled Natural Languages (CNL)s offer an incentive to the novice user to annotate, while simultaneously authoring, his/her respective documents in a user-friendly manner,yet shielding him/her from the underlying complex knowledge representation formalisms. CNLs have already been successfully applied within the context of ontology authoring, yet very little research has focused on CNLs for semantic annotation. We describe a user friendly semantic annotator, based on Controlled Language for Information Extraction (CLIE) tools, which permits non-expert users to semi-automatically both author and annotate meeting minutes and status reports using controlled natural language.

2 Controlled Natural Languages and Semantic Annotation

"Controlled Natural Languages are subsets of natural language whose grammars and dictionaries have been restricted in order to reduce or eliminate both ambiguity and complexity.[3]" The use of CNLs for ontology authoring and population is by no means a new concept and it has already evolved into quite an active research area[4]. A natural overlap exists between tools used for both ontology

[3] http://www.ics.mq.edu.au/~rolfs/controlled-natural-languages/

creation and semantic annotation, for instance the CLIE technology permits ontology creation and population by mapping both concept definitions and instances of concepts to a ontological representation using CLOnE - Controlled Language for Ontology Editing[1]. However, there is a subtle difference between the process of ontology creation and population and that of semantic annotation. We describe semantic annotation as "a process as well as the outcome of the process. Hence it describes i) the process of addition of semantic data or metadata to the content given an agreed ontology and ii) it describes the semantic data or metadata itself as a result of this process"[3]. Of particular importance here is the notion of the addition or association of semantic data or metadata to *content* - in this context a semantic note on the Semantic Desktop. As with any annotation environment, a major drawback is that in order to create metadata about a document, the author must *first* create the content and *second* annotate the content, in an additional a posteriori, annotation step. In the context of our annotator we seek to merge both authoring and annotation steps into one. Consequently, the user authors parts of his/her notes in CNL while *simultaneously* creating relation metadata to describe its content. Very little research is available with respect to CNLs for semantic annotation. For instance, Project HALO[4] was a research venture sponsored by Vulcan Inc[5]. It aimed to develop, a "Digital Aristotle"- a comprehensive, automated tutor and research assistant.A CNL for semantic annotation was implemented as part of the project, yet no public material describing the CNL is available for scientific scrutiny.

3 A Use Case for Controlled Natural Language for Semantic Annotation

CNLs cannot offer a panacea for semi-automatic annotation since it is unrealistic to expect users to annotate every textual resource using CNL, however there are certain use-cases where CNLs can offer an attractive alternative as a means for semi-automatic semantic annotation, particularly in contexts, where controlled vocabulary or terminology is implicit such as health care patient records or business vocabulary. Our use case focuses on administrative tasks such taking minutes during a project team meeting and weekly status reports. Very often such note taking tasks can be repetitive and boring. In our scenario the user is a member of a research group which in turn is part of an integrated EU research project. Based on pre-defined templates, the user *simultaneously authors and annotates* his/her meeting minutes or status reports in CNL, using a semantic note taking tool - SemNotes[6], which is an application available for Nepomuk-KDE[7] - the KDE instance of the Social Semantic Desktop. The metadata is available for immediate use after creation for querying and aggregation, whereby

[4] http://www.projecthalo.com/
[5] http://www.vulcan.com
[6] http://smile.deri.ie/projects/semn
[7] http://nepomuk.kde.org/

the retrieved RDF triples can be passed to a Natural Language Generator to produce tailored textual reports and summaries.

4 Implementation

In our scenario, the CNL annotator is realised within a Semantic Note. The CNL is anchored to existing semi-structured data such as a `AgendaTitle`, `Scribe` or `ActionItem` based on predefined meeting minutes or status report templates. The annotator is based on CLIE. The CNL itself is very similar to the CLOnE language with significant modifications. The annotator architecture contains a standard GATE pipeline[8](see Figure 1) which includes the following language processing resources: The GATE English tokeniser, the Hepple POS tagger, a morphological analyser, a gazetteer list component for recognising useful key-phrases, such as structured elements from the templates and reserved CNL phrases. Any sentences for example, preceded by a `Comment:` element are considered candidates for controlled language parsing. Any remaining tokens from the CNL sentence which are not recognised as reserved CNL key-phrases are used as names to generate links to ontological objects(See Figure 2). This is followed by a standard Named Entity(NE) transducer in order to recognise useful NEs, a `preprocessing` JAPE[9] finite state transducer(FST) for identifying quoted strings, chunking Noun Phrases(NPs) and additional preprocessing. A second gazetteer list look up is applied which identifies trigger phrases associated with NEs which intersect with quoted and unquoted NP annotation spans. Additional feature values are then added to the NP chunks to indicate the appropriate class to link an NP chunk as an instance to. The last FST parses the CNL from the text and generates the metadata. The current tool is bootstrapped via the Nepomuk Core Ontologies[10] and currently the application creates/populates a meeting minutes/status report ontology which references the users Personal Information Model Ontology(PIMO) [11], via the GATE Ontology API. We intend to modify the code to write directly to Nepomuk KDE RDF store.

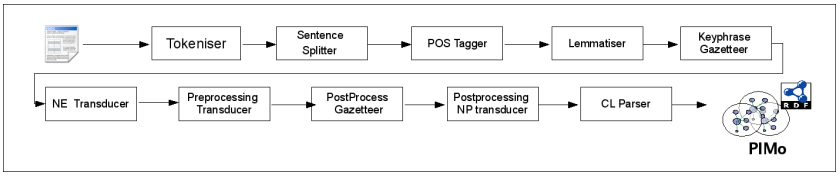

Fig. 1. The CNL Semantic Annotator pipeline

[8] General Architecture for Text Engineering, See http://gate.ac.uk/
[9] Java Annotations Pattern Engine
[10] http://www.semanticdesktop.org/ontologies/
[11] http://www.semanticdesktop.org/ontologies/2007/11/01/pimo/

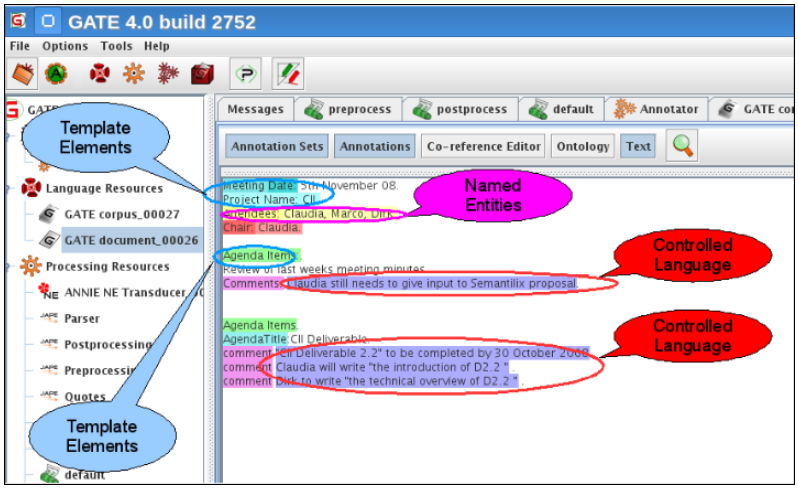

Fig. 2. CNL Annotator visualised in GATE

5 Conclusion

Incentive for the user to annotate his/her respective documents plays an important role for the realisation of both the Semantic Web and Social Semantic Desktop. We have described a Semantic Annotator which allows non expert users to simultaneously create content within, and add relational metadata to, notes on the Semantic Desktop, using CNL. Furthermore, our annotator has already been implemented and wrapped as a plugin for a semantic note taking tool. Finally, we intend to complete the integration with Nepomuk KDE and evaluate the user-friendliness of our annotator based on the previously successful empirical methods employed in CLOnE [1].

References

1. Brian Davis, Ahmad Ali Iqbal, Adam Funk, Valentin Tablan, Kalina Bontcheva, Hamish Cunningham, and Siegfried Handschuh. Roundtrip ontology authoring. In Amit P. Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy W. Finin, and Krishnaprasad Thirunarayan, editors, *International Semantic Web Conference*, volume 5318 of *Lecture Notes in Computer Science*, pages 50–65. Springer, 2008.
2. S. Decker. The social semantic desktop: Next generation collaboration infrastructure. *Information Services and Use*, 26(2), 2006.
3. Siegfried Handschuh. *Creating Ontology-based Metadata by Annotation for the Semantic Web*. PhD thesis, 2005.
4. P. R. Smart. Controlled natural languages and the semantic web. Technical report, School of Electronics and Computer Science, University of Southampton, 2008,(Unpublished).