A Use case for Controlled Languages in Ontology based Knowledge Management

Pradeep Varma Dantuluri

August 2010

Contents

List of Figures

List of Tables

Declaration and Disclaimer

I declare that this thesis is composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified. The work reported in this thesis is part of research conducted in DERI, Galway within the DERI Líon-2 project. Pradeep Varma Dantuluri

Acknowledgements

Abstract

Although the Semantic web is steadily gaining in popularity, it remains a mystery to a large percentage of Internet users. This can be attributed to the complexity of the technologies that form its core. Creating intuitive interfaces which completely abstract the technologies underneath, is one way to solve this problem. A contrasting approach is to ease the user into understanding the technologies. We propose a solution which anchors on using controlled languages as interfaces to semantic web applications. This paper describes one such approach for the domain of meeting minutes, status reports and other project specific documents. A controlled language is developed along with an ontology to handle semi-automatic knowledge extraction. The contributions of this paper include an ontology designed for the domain of meeting minutes and status reports, and a controlled language grammar tailored for the above domain to perform the semi-automatic knowledge acquisition and generate RDF triples. This paper also describes two grammar prototypes, which were developed and evaluated prior to the development of the final grammar, as well as the Link grammar, which was the grammar formalism of choice.

Chapter 1

Introduction

1.1 Motivation

The Semantic web[1] aims to simplify the process of building knowledge-based applications by enabling a web of inter-operable and machine-readable data. This is done by formalizing the descriptions of the structure and semantics of the data available on the web. The linked data initiative[2] is a positive step in that direction, exposing huge amounts of data for further analysis and use by other applications. However creating and exposing linked data is a task that requires thorough knowledge of various technologies which would be a huge hassle for the novice users. A solution to this is to create technologies which would enable the average internet user to annotate and embed data in his/her own textual resources. This work aims to explore the possibility of using controlled natural languages(hereby referred to as CNL) as an interface for semantic annotation, specifically targeting the domain of project documents like meeting minutes, status reports, etc. The major goal was to enable novice users to author and annotate text documents simultaneously using a controlled language. Furthermore, these documents can be parsed to extract the implicit knowledge contained, due to the enforcement of a fixed grammar and vocabulary. This completes the task of converting human-readable controlled language texts to machine-interpretable structured information which could be further exploited. Previously this approach was used to build an annotation tool along with prototypes of the grammar and ontologies for the meeting minutes domain [2].

1.2 Research Goal

Using CNLs as interfaces to knowledge acquisition applications, soothens the barriers of entry for novice users thereby leading to a greater public involvement.

The main contributions of the thesis include

- An Annotation Software platform for the knowledge acquisition and management pertaining to the meeting minutes domain

[1] `http://www.w3.org/2001/sw/`
[2] `http://linkeddata.org/`

- The development and publishing of an ontology which models the domain of project documents
- Development of the CLANN grammar using link grammars and the corresponding parser
- A GWT based web interface for the novice users to aid them in authoring texts in Controlled language

1.3 Thesis Layout

Chapter 2

Related Work

The proposed work touches upon a combination of topics ranging from CNLs and using them as an interface for Knowledge aquisition, Development and maintainence of ontologies, the Link grammar formalism, and finally Semantic Web and the Linked data . Essentially the work aims to create a smart application which would allow users to write meeting minutes and status reports in a controlled and restricted version of the english language. This controlled language is machine processable, hence parsed, understood and relevant information is extracted. This information is formatted as RDF data (RDF tripples) using a pre-defined ontology and made accesible through linked data standards.

2.1 Semantic Web and Semantic Wikis

2.1.1 Semantic Web

2.1.2 Linked Data web

2.1.3 Wikis and Semantic Wikis

Other related work involves the application of Controlled Languages for Ontology/Knowledge base querying, which represent a different task than that of knowledge creation and editing but are worth mentioning for completeness sake. Most notably *AquaLog*[1] is an ontology-driven,portable question-answering (QA) system designed to provide a natural language query interface to semantic markup stored in a knowledge base. PowerAqua [**?**] extends AquaLog, allowing for an open domain question-answering for the semantic web. The system dynamically locates and combines information from multiple domains. Write about Semnatic Web. Write about Linked Data. Explain the reach and progress of Linked data and semantic web with examples. The proposed work aims to use Linked data technologies to open up the data of the application and hence connecting to the huge amount of knowledge already availiable on the internet. This enables the users to efficiently exploit the vast amount of information according to his need.

[1] `http://kmi.open.ac.uk/technologies/aqualog/`

2.2 Ontologies and Ontology Engineering

2.2.1 Overview of ontologies

2.2.2 Ontology languages

2.2.3 Ontology Engineering

Methontology

Explain ontologies and their need. Explain a few of the methodologies to maintain and develop ontologies. The domain of meeting minutes and status reports is modelled as the PDO ontology. Further details are described in the section below.

2.3 Controlled Languages and the Semantic Annotation

2.3.1 Controlled Language interfaces for HLT

"Controlled Natural Languages (CNL)s are subsets of natural language whose grammars and dictionaries have been restricted in order to reduce or eliminate both ambiguity and complexity"[6]. CNLs were later developed specifically for computational treatment and have subsequently evolved into many variations and flavours such as Smart's Plain English Program (PEP) [**?**], White's International Language for Serving and Maintenance (ILSAM) [**?**] and Simplified English[2]. They have also found favour in large multi-national corporations, usually within the context of machine translation and machine-aided translation of user documentation [6, **?**]. The application of CNLs for ontology authoring and instance population is an active research area. *Attempto Controlled English*[3] (ACE) [**?**], is a popular CNL for ontology authoring. It is a subset of standard English designed for knowledge representation and technical specifications, and is constrained to be unambiguously machine-readable into DRS - Discourse Representation Structure. , a form of first-order logic. (It can also be re-targeted to other formal languages.) [**?**]. The Attempto Parsing Engine (APE) consists principally of a definite clause grammar, augmented with features and inheritance and written in Prolog [**?**]. ACE OWL, a sublanguage of ACE, proposes a means of writing formal, simultaneously human- and machine-readable summaries of scientific papers [**?**, **?**].

2.3.2 Controled Languages for Knowledge Management

The Rabbit CNL is a another well known CNL[**?**]. It is similar to CLOnE in its implementation but is much more powerful with respect to ontology authoring capabilities and expressivity. It has also been favorably evaluated by users in the Ordinance Survey domain but is targeted towards ontology authoring and not semantic annotations. It has been integrated into Semantic Media Wiki,

[2] `http://www.simplifiedenglish\-aecma.org/Simplified\_English.htm`
[3] `http://www.ifi.unizh.ch/attempto/`

the purpose of which to create a user friendly collaborative ontology authoring using multiple CNLs[**?**].

Other related work (in that creates A-box statements) is WYSIWYM (*What you see is what you meant*)[**?**]. It involves direct knowledge editing with natural language directed feedback. A domain expert can edit a knowledge based reliably by interacting with natural language menu choices and the subsequently generated feedback, which can then be extended or re-edited using the menu options. However this differs substantially from semantic annotation. Similar to WYSIWYM is *GINO* (Guided Input Natural Language Ontology Editor) provides a guided, controlled NLI (natural language interface) for domain-independent ontology editing for the Semantic Web. GINO incrementally parses the input not only to warn the user as soon as possible about errors but also to offer the user (through the GUI) suggested completions of words and sentences—similarly to the"code assist" feature of Eclipse[4] and other development environments. GINO translates the completed sentence into triples (for altering the ontology) or SPARQL[5] queries and passes them to the Jena Semantic Web framework. Although the guided interface facilitates input, the sentences are quite verbose and do not allow for aggregation. Static grammar rules exist for the controlled language but in addition, dynamic grammar rules are generated from the Ontology itself as an amendment of additional parsing rules to GINO's grammar in order to guide the user. This permits the system to handle a domain shift, however this is heavily dependent on any linguistic data or RDF label data encoded the ontology [**?**].

Finally, [**?**] presents an Ontology based Controlled Natural Language Editor, similar to GINO, which uses a CFG (Context-free grammar) with lexical dependencies - CFG-DL to generate RDF triples. To our knowledge the system ports only to RDF and does not cater for other Ontology languages.

2.3.3 Semantic Annotation

Different approaches (focus on Semi automatic + manual)

While there are a plethora of tools for manual and (semi-)automatic tools semantic annotation tools (which apply knowledge based approaches using applied NLP ,machine learning techniques or both to the process), to our knowledge, very little research exists involving the application of controlled natural language to semantic annotation.

" A Controlled Natural Language (CNL) is a subset of a natural language whose grammar and vocabulary has been restricted in order to reduce or eliminate ambiguity and complexity"[6]. CNLs have been successfully applied as natural language interfaces to enable users to communicate with the application easily without undergoing any rigourous training.

CNLs have already been applied to ontology authoring and population [4]. Previous work by the authors [2] tackle the problem of applying CNLs to semantic annotation. The process of semantic annotation according to [5] involves addition or association of semantic data or meta-data to the content, according to an agreed-upon ontology. Conventionally, work on semantic annotation focused on two-step approaches where the authoring of a document has to precede

[4] `http://www.eclipse.org/`

[5] `http://www.w3.org/TR/rdf-sparql-query/`

the annotation of the same. This problem can be overcome by adopting a *latent* annotation[2] approach by the use of controlled languages, which merges both the authoring and annotation steps into one. The information is encoded in the restricted vocabulary and grammatical structure of the controlled language. However, preliminary evaluations suggest that annotating every piece of information using a CNL makes the task quite verbose, thereby demotivating the users. Our previous work explored a simple solution to this by supplementing the CNL using templates which encode implicit domain information.

2.4 Grammar formalisms

Grammar of a particular language is a list of principles and rules which direct the placement of words to form meaningful sentences of the the language. Grammars of natural langauges have been studied extensively over the past decades, and various formalisms have been defined. They can be broadly categorised into constituent and dependency based formalisms. The fundamental idea of constituency grammars is, words can be grouped into meaningful units or phrases. Context-free grammars (CFG) are the most widely used constituent formalism, described by [1]. CFGs consist of a set of rules representing the grammar of a language, which usually recognize legitimate sentences of the language and generate a tree-like parse structure, breaking the sentence into meaningful phrases. CFGs are better suited to work with the the syntactic knowledge that can be modelled by grammars, hence forming a backbone of our understanding of the syntax of natural languages. Dependency grammars, however, centre around the relations between various words in a sentence. The syntactic structure of a sentence is describes in terms of words and various kinds of relations among them, thereby building relations between words instead of genrating a tree-like structure. Link grammars, described by [7], are a special kind of dependency grammars where the links/relations are directional along with an added condition where each word should be linked to atleast one other word.

A Link grammar consists of a set of words along with linking requirements of each word. The requirements also encode the directionality for each link. A sentence is said to be part of the grammar if each word in the sentence is linked to atleast one word, while satifying the requirements for each link. Furthermore, the links should not overlap. A more detailed explanation of the Link grammar is given in Sections below.

Our work focusses on extracting meaningful tripples from the input controlled language. This approach places a high importance on the links between various words of a sentence. Hence we preferred a dependency based grammar instead of the usual constituent grammars. Why prefer this? elaborate... Moreover, link grammars are known to work especially well if the lexicon of a grammar is fixed. who says so? reference.. Since we aimed to design a controlled language for the domain of meeting minutes and status reports, our lexicon is limited, hence justifying the choice of Link grammars.

Additionally we aim to build a guided input CNL editor which should be able to parse at every step and return a list of possible words that can act as suggestions. An link parsers can be adapted to do so, as they dont need a complete parse for any given sentence.

Perhaps the most closely related technology is the Semantic MediaWiki tech-

nology [6] have become a popular way of adding semantics to user generated Wiki pages. A traditional wiki creates links between pages without defining the kind of linkage between pages. Semantic MediaWiki allows a user to define the links semantically, thereby adding meaning to links between pages. Each concept or an instance has a page in Semantic Wiki, and each outgoing links from this page is annotated with well-defined properties as links. However this kind of approach is not suitable to the kind of semantic annotation that we aim for. The Semantic Media Wiki model forces the users to use the wiki pages for content creation and to create a new page for each instance but does not offer a method to annotate arbitrary text documents which are not intended to be used as wiki pages. Moreover, the relational metadata represented in a Semantic Media Wiki always has the corresponding page as its subject, thereby restricting the creation and description of other relevant entities.

[6] More information about Semantic MediaWiki can be found at http://semantic-mediawiki.org/wiki/Help:Introduction_to_Semantic_MediaWiki *as accessed on 21/06/09*

Chapter 3

Methodology

3.1 CLANN

3.1.1 Definition

CLANN (Controlled Language for ANNotation)[1] builds on the experiences gained from the previous work, by incorporating redesigned versions of the grammar and the domain ontology. CLANN is designed to be an end-to-end semantic web application complete with a domain ontology, a persistent layer based on RDF and a user interface for editing and authoring documents. The domain was expanded to include all the documents in a project specific setting (for example, meeting minutes, status reports, etc).

3.1.2 Use case

3.2 PDO Ontology

The domain of meeting minutes and status reports was used to engineer an ontology for the purpose of knowledge management. The initial CLANN prototype was bootstrapped using the the nepomuk ontologies[2] and later extended by the MEMO ontology. However , the MEMO ontology was only used as a proof of concept implementation of the domain. Later, this was completely redesigned and a new ontology PDO (Project Document ontology) was developed in accordance with proper ontology design principles, specifically the Methontology[3] approach. The PDO ontology, described using RDFS and OWL-DL, models the inherent structure and concepts of various documents in a project-specific setting, like meeting minutes, status reports etc. A pictorial representation of the main aspects of the ontology is shown in Figure **??**. [3]

[1] In this document *CLANN* refers to the annotation software platform as well as the grammar

[2] `www.semanticdesktop.org/ontologies/`

[3] for a complete specification of the PDO ontology please refer to `http://ontologies.smile.deri.ie/pdo`

3.2.1 Design of the ontology

3.2.2 Development and publishing the ontology

3.3 CLANN Grammar

3.3.1 Grammar formalisms

Various grammar formalisms have been used over the years for understanding natural language. Phrase structure grammars (PSG), the most widely used formalism, model the inherent structure of the sentences of a language by breaking it into different phrases. They belong to the class of generative grammars and are composed of a set of productions or rules which break-up the sentences into meaningful phrases. Dependency grammars(DG), however, concentrate on the links between words without paying attention to the word order. Structure of a sentence is not broken down into phrases, but determined by adding relations between a head word and its dependent words. There have been many variations of gramamar formalisms that stemmed out of both PSGs and DGs. The next few sections describe one such variation of the dependency grammar, the Link grammar, and justifies its selection.

3.3.2 Link Grammar

Link grammars, introduced by [**?**], are a variation of dependency grammars. Similar to DGs, the link grammars use realtions between words to generate a structure for a sentence. However, unlike DGs, the links also encode information about directionality and distance. Moreover, they do not enforce a head-dependent relationship like the DGs.

[**?**] defines link grammar as follows:

> *A sequence of words is a sentence of the language if there is a way to draw links between words in such a way that*
>
> - *the linking requirements of all the words are satisfied,*
> - *the links do not cross, and*
> - *the words form a connected graph*

The linking requirements of each word are specified as a dictionary, which forms the basis of the link grammar. Each entry in the dictionary consistes of a word or a group of words belonging to the same grammatical category, appended on the right-hand-side with its linking requirements. The linking requirements are a series of connectors joined by the logical operators *&* and *or*. Each connector denotes the type and direction of the link. It is a label followed by *+/-* . *+* denotes a link to the right and - denotes a link to the left. For illustration purposes, an example of a sentence parsed using a very simple link grammar is provided in Figure 3.1, and an explanation of the same is provided below.

The *D+* connector on the word *the* denotes that *the* is expecting a *D* link to its right. So It can connect to any word which has a *D-* connector, which, in this case, is either *boy* or *apple*. The word *ate* has an *&* operand on *S-* and *O+*. This means, for the word *ate* to be part of a valid sentence, it should connect to

words	linking requirements
the	D+
small	A+
ate	S- & O+
boy apple	(A- & D- & S+) or (D- & O-)

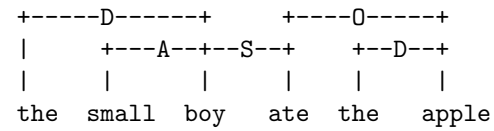

Figure 3.1: A sample Link grammar and parse structure.

both an S connector to its left and an O connector to its right. The case for the nouns *boy* and *apple* is more interesting. They have two expressions joined by the *or* operand. On closer observation, the first one, *(A- & D- & S+)*, models the behavior of a subject noun and the second one, *(D- & O-)*, models that of an object noun. The reader should also note that the order of the connectors is also valuable. The expression *(A- & D- & S+)* also declares the order of linking. So an A link should be made to a word closer than the D link. This is illustrated in the parse structure shown in Figure 3.1.

3.3.3 Why Link Grammar?

The main design principles for the CLANN grammar are ease of use and the ability to extend the ontology. However, the develpoment of the grammar posed different challenges.

One major priority was to extract RDF tripples from the sentences. This works very well with the link grammar parse, because the the tripples can directly be extracted by mapping the links. In the example shown in Figure 3.1, the tripple *boy ate apple*, can be easily extracted from the left and right links of the word *ate*. This is not the case with phrase structure grammars, where extracting dependencies requires detailed ananlysis of the tree structure.

Another major priority was to develop an intelligent editor on top of the grammar, which supports auto-suggestion and sentence-completion. An intuitive editor which assists the user while writing the CNL sentences, goes a long way in helping him to quickly learn the restrictions of the grammar. This requires an ability to predict text and check the grammatical correctness of partial sentences. The dictionaries of the link grammar provide valuable information about all the words of the language, which can be exploited for the purpose.

3.3.4 Designing the CLANN grammar

CLANN is designed to enable representing any information that cannot be captured by the templates. To better explore applicability, two independent prototypes of the grammar CLANN1 and CLANN2 were developed[ref for CNL09]focusing alternatively on usability and expressivity. This work has eventually led to

CLANN3, which is essentially a merge between the former two grammars, incorporating most of the advantages, albeit a few changes.

CLANN1 prototype

CLANN1 is designed with a major focus on usability. Each sentence adhered to one of the syntactic rules and used a very lenient vocabulary. This domain vocabulary was derived by corpus analysis using Word Smith tools [4] on the document corpus. This ensured that most of the sentences resembled normal english sentences. CLANN1 is grammatically lax in comparision to typical CNL approaches to knowledge creation. A modified shallow parser is then used to extract the knowledge and instantiate the ontology.

CLANN2 prototype

CLANN2 was designed with a major focus on expressivity. It differs from the conventional notion of CNL, whereby the entire document is written in CNL, rather it allows the user to add snippets of CNL text, enclosed in `"[ ]"`, to the document or associate them to a particular text in the document.These snippets should adhere to a *subject-verb-object* syntax, where the subject is either specified in the snippet or taken from the free text. The vocabulary for CLANN2 also includes the vocabulary of the ontology, thereby allowing the user to represent any kind of relational meta-data. This approach was inspired by the CLOnE Language [4].

The finer details of design and implementation of these grammars are described in [2]. Both grammars use a common template, described above, which is initially parsed to extract the inherent meta data of the document (in this case, meeting minutes).

CLANN grammar

The CLANN grammar is essentially a combination of CLANN1 and CLANN2, incorporating the snippets of CLANN2 into controlled text of CLANN1 instead of free text. For other sample documents and information please refer to the project home page[5].

Templates

Inorder to supplement the CLANN grammar, we templates which encode implicit domain information, an example of which is shown below.

[4] `http://www.lexically.net/wordsmith/version5/index.html`
[5] `http://smile.deri.ie/projects/clann`

```
Project Name:  <String>
Group Name:  <String>
Date:  <Date>
Chair:  <String>
Attendees:  (<String>)+
Scribe:  <String>
Action Item:<String>:<String>(:<String>)?
(<CNL>)+
Agenda:  <String> (<CNL>)+
Poll:<String>:<String> (<String>:<String>)+
```

These templates were constructed by analysing a collection of in-house meeting minutes and status reports of the Nepomuk project[6]. This approach combines the benefits of the two by using templates for mundane information annotation and CNL for other non-mundane information, consequently minimising the effort and enhancing the user experience.

3.4 Design and development of CLANN

3.4.1 CLANN Annotation Platform

3.4.2 Design of the platfrom

3.4.3 Architechture

3.4.4 Modules

3.4.5 Interfaces availiable

3.4.6 Sceenshots

3.4.7 The CLANN grammar

[6] http://dev.nepomuk.semanticdesktop.org/wiki/WikiStart

Chapter 4

Evaluation

4.1 Experimental Evaluation

4.1.1 Methodology

4.1.2 Sample quality

4.1.3 Discussion

4.1.4 User feedback

Chapter 5

Conclusions and Future Work

5.1 Conclusion

5.1.1 Research problem revisited

5.1.2 Further Research

5.1.3 Conclusion

Bibliography

[1] Noam Chomsky. Three models for the description of language. *IRE Transactions on Information Theory*, 2:113–124, 1956. `http://www.chomsky.info/articles/195609--.pdf` – last visited 14^{th} January 2009.

[2] Brian Davis, Pradeep Varma, Siegfried Handschuh, Laura Dragan, and Hamish Cunningham. On designing controlled natural languages for semantic annotation, 2009.

[3] Mariano Fernandez-Lopez, Asuncion Gomez-Perez, and Natalia Juristo. Methontology: from ontological art towards ontological engineering. In *Proceedings of the AAAI97 Spring Symposium*, pages 33–40, Stanford, USA, March 1997.

[4] A. Funk, V. Tablan, K. Bontcheva, H. Cunningham, B. Davis, and S. Handschuh. Clone: Controlled language for ontology editing. In *ISWC/ASWC*, pages 142–155, 2007.

[5] Siegfried Handschuh. *Creating Ontology-based Metadata by Annotation for the Semantic Web*. PhD thesis, 2005.

[6] Rolf Schwitter. Controlled natural languages.

[7] Daniel D. K. Sleator, C Fl Daniel Sleator, and Davy Temperley. Parsing english with a link grammar. 1991.