A Pilot Experiment in Knowledge Authoring as Dialogue[*]

Artemis Parvizi[†] Caroline Jay[‡] Christopher Mellish[†] Jeff Z. Pan[†] Yuan Ren[†]
Robert Stevens[‡] Kees van Deemter[†]

Abstract

This project aims to build an ontology authoring interface in which the user is engaged in a dialogue with the system in controlled natural language. To investigate what such a dialogue might be like, a layered annotation scheme is being developed for interactions between ontology authors and the Protégé ontology authoring environment. A pilot experiment has been conducted with ontology authors, which reveals the complexity of mapping between user-interface actions and acts that appear in natural language dialogues; it also suggests the addition of some unanticipated types of dialogue acts and points the way to some possible enhancements of the authoring interface.

1 Introduction

Ontology authoring – the process of creating and modifying an ontology in order to capture the knowledge about a domain – is hard. Studies such as (Rector et al., 2004) and (Dzbor et al., 2006), for example, have shown that ontology authors frequently misunderstand axioms in the ontology. These studies also suggest that current ontology authoring tools fail to support some of the actions that authors would like to perform, and that users would like to be warned against potential mis-uses of axioms and unforeseen consequences of those axioms.

This paper reports on work in progress, in which we study the interaction of human users with an ontology authoring interface, with the ultimate aim of developing a tool that will permit a much richer knowledge authoring experience, thereby addressing the above-mentioned problems associated with existing knowledge authoring. We envisage developing a plugin to the knowledge authoring interface Protégé[1] that will allow authors to interact with the ontology via a controlled natural language (CNL) dialogue that offers users some of the freedom of natural language without causing insurmountable natural language understanding problems to the system. Specifically, we report on a pilot experiment in which we observed human editors who were invited to use Protégé while talking to an experimenter, and on a layered annotation scheme that we are developing for annotating the resulting interactions. Once a stable annotation scheme has been reached, we shall use the scheme to obtain annotations involving a larger number of users and a larger number of knowledge authoring tasks, the results of which will inform the design of the new knowledge authoring interface. In this paper, we focus on the following questions:

- If knowledge authoring is viewed as a dialogue, what would be the main moves that one would expect to see in these dialogues?
- How are these dialogue moves grounded in lower-level actions like "Finding subsumers for a given concept", or "Looking at a concept"?
- How does the annotation of a knowledge authoring dialogue compare to the annotation of real spoken dialogue?

[*] This work was supported by EPSRC grants EP/J014354/1 and EP/J014176/1.

[†] University of Aberdeen, UK

[‡] University of Manchester, UK

[1] http://protege.stanford.edu

Ontology authoring aided by CNL has been addressed from various points. ACE (Fuchs et al., 1999) and PENG (Schwitter, 2002) allow users to express specifications that can be translated into an unambiguous logical language, understandable to machines. CLCE (Sowa, 2004) resembles ACE but, being closer to English, its output is more readable. SWAT (Power, 2009) uses natural language generation to enable users to produce description logic statements. Similar to CLCE, (Bernardi et al., 2012)'s work is another example of a novel controlled language capable of assisting ontology authoring.

Whereas this previous work has addressed the problem of producing isolated utterances relevant to knowledge authoring using CNL, this project attempts to further develop knowledge authoring by adopting new interactive methods inspired by human dialogue. For example, we hypothesise that by allowing an author to pose *what-if* questions prior to authoring an axiom in an ontology, the authoring process will run in a more informed manner. The following dialogue is an example of the type of communication that this project is attempting to make possible during knowledge authoring:

U1 I want a Moral Right to be able to have a date.
S1 Moral Rights currently have no date property.
U2 What if I import timeframe.owl?
S2 That is consistent. There is now a date property.
U3 What if a Moral Right must have exactly one date?
S3 That implies that Attribution Rights and Integrity Rights must have exactly one date.

2 A new dialogue annotation scheme for Knowledge Authoring

We decided to use a layered model of human-computer interaction to analyse the interactions between knowledge editors and Protégé. This general idea has a long pedigree. Moran (1981) proposed a command language grammar, Foley et al. (1982) suggested a 4-layer design model, Buxton (1983) expanded Foley et al.'s model into a 5-layer design model, and Nielsen (1986) presented a virtual communication protocol. Generally, the aim is to have a better understanding of various aspects of human machine interaction, recognising that a high-level action may be "implemented" in various ways. In this section, we sketch our intial three-level scheme for analysing the interaction between users and Protégé, which was then modified in light of our experiment.

KLM level. The lowest level of analysis is akin to the Keystroke Level Model (KLM) (Card et al., 1986). This level records the physical interactions of the user with the system. KLM level includes acts such as clicks, mouse hovers, and eye movements.

Authoring level. At a higher level of analysis, we wanted to record what it is that users attempt to do and the high-level actions that they perform to accommodate these goals. Some existing analyses (Goncalves et al., 2011; Abgaz et al., 2012) have attempted to understand and categorise changes to the knowledge base during its development. The intention of the present work is not only to categorise such change, but also to understand users' thinking and, ultimately, to work towards a greater knowledge authoring functionality. We decided to group authoring-level acts into general acts and interface acts. *General acts* are grouped into (1) various types of observations (for example, looking whether a class of a given name exists in the ontology; looking to see what instances of a class exist), (2) addition, updating, and deletion of axioms, (3) noticing and resolving inconsistencies. *Interface acts* include (1) common interface features such as undoing or redoing a previous action, and (2) seeking explanations of inconsistencies.

Dialogue level. Annotation at this level involves a creative step where actions are analysed *as if* they were instances of acts in a natural language dialogue. These dialogue acts will provide basis for the design of an interface for knowledge authoring that is similar to natural dialogue. Annotation of dialogue acts is standard in the analysis of Natural Language dialogues (Bunt, 1994; Ginzburg, 1994), but this is not yet used in ontology authoring. Figures 1 and 2 show our initial intuition of dialogue acts, which are categorised into user and system's dialogue acts.

- *Informative Questions* The aim of this type of question is to acquire some information, for example, "Where does class X appear in the classification hierarchy?". The system is intended to search

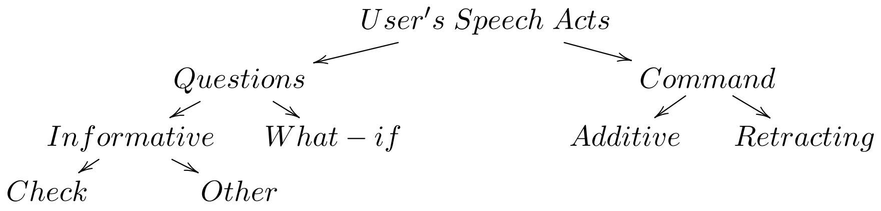

Figure 1: User's Dialogue Acts

through the inferred hierarchy for a response.

- *What-if Questions* Based on the current context, this type of question informs the user about a possible future state of the system. The system needs to perform an action and report its consequences. However, the decision to apply the change is made by the user later.
- *Check Questions* Similar in nature to informative questions, check questions provide answers to user's inquiries, but where the user has a strong expectation about the answer.
- *Additive Command* Informing the system of a certain statement that has to be added or made true in the ontology. The consistency of the ontology needs to be checked and reported to the user. The decision how and when to address the inconsistency has to be made by the user later.
- *Retracting Command* In certain circumstances when the ontology has become inconsistent, the user can either ask an informative question, or a check question and based on the answer decide to retract a change; or, the user can simply retract one or more of the uttered statements.

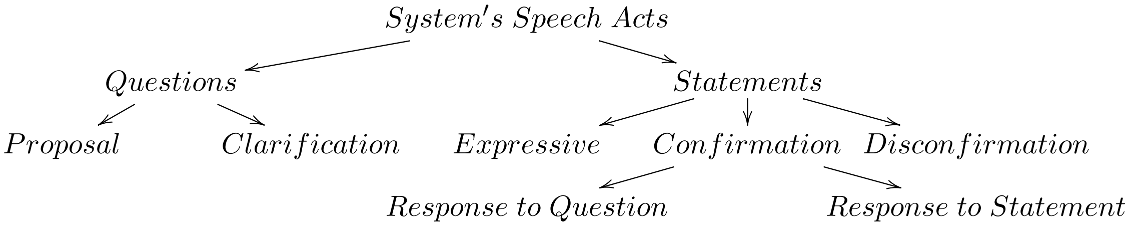

Figure 2: System's Dialogue Acts

- *Expressive Statements* A response to an information seeking or a *what-if* question. Based on the type of question to be addressed, the system may or may not be involved in an authoring process.
- *Confirmation Statements in Response to Questions* Informing the user that an authoring act had the desired effect.
- *Disconfirmation Statements in Response to Question* Informing the user of the consequences to an authoring act, which may not be desired.
- *Confirmation Statements in Response to Statements* Confirmation of a successful authoring act.
- *Proposal Questions* A suggestion by the system to undertake an action.
- *Clarification Questions* A question posed to the user to clarify a command.

3 A Pilot Experiment

The 5 participants in the pilot study were computer scientists who were experienced users of OWL and Protégé. They were asked to set up the Protégé interface as they normally operated it. They were asked to explore the People and Pet ontology acquired from the Tones repository [2], and manipulate it as they saw fit. They were asked to follow the *think aloud* protocol, and describe their moves. In addition, their interactions and eye movements were video-recorded. Where the experimenters were unclear about a description of what was happening, they asked the subject for clarification. Occasionally, when the

[2] http://owl.cs.manchester.ac.uk/repository/

subject felt unsure how to proceed, the experimenter suggested an action (for example, Why don’t you look at the axioms now appearing in red?”)

Table 1: A pilot experiment and various levels of annotation

	Speech/Action	**Authoring level**	**Dialogue Level**
7:35	“he’s a car driver”	Adopt goal 1: (Mick is a car driver)	Expression of intention
⋮	⋮	⋮	⋮
8:57	“make the … thing that he’s driving … ”	Adopt goal 2: (Q123 is a car)	Expression of intention
9:01	“I will just remove whatever was there and … ”	Adopt goal 3: (Q123 is only a car)	Expression of intention
9:03	“oh I know!”	Abandon goal 3	Retraction of intention
9:03	clicks ‘Types+’ in ‘Description Q123’	Add axiom: (Q123 is a car)	Additive command
9:04	“add car as well”		
9:04	types ‘car’ in ‘Q123’ dialogue		
9:05	“and see whether it’s invalid”	Check: (Mick is a car driver)	Check Question
9:08	runs reasoner		
9:08	“if it is really the same thing then it should be disjoint”	Notice: (vehicle type not disjoint)	–
		Note goal: (vehicle types disjoint)	Expression of desire
9:11	“mm-hm”	Confirm check	Confirmation statement
9:16	clicks ’Mick’ in ‘individuals’ window	Goal achieved: (Mick is a car driver)	Retraction of intention

Knowledge authoring sessions were annotated with reference to the scheme outlined above but, crucially, annotators wrote annotations in a free style: the interface did not limit them to a fixed set of labels implied by our initial annotation scheme. A small annotated segment of one session is presented in Table 1. In the speech/action column, speech refers to the user’s comments and is displayed within quotations, and actions refers to the physical interactions between the user and the Protégé interface.

Our initial intuitions about dialogue acts were shown to be far from perfect. Perhaps unsurprisingly, it became clear that the mapping between KLM, Protégé and dialogue levels can be very complex. An annotation in one layer can correspond to more than one in another layer. For example, a subject may observe the definition or the location in the hierarchy of a concept for various *conceptual reasons*, for instance to check consistency (a *Check Question* in our scheme) or to prepare for adding a new concept (an *Additive Command*). Difficulties of a similar kind are well known from the annotation of natural language dialogue, for instance whether a given spoken utterance (for example, “You are 29 years old”) has declarative or interrogative meaning (Beun, 1989). It also became clear that a separate category is required where a command is both Additive and Retracting at the same time, such as a *Modifying Command*. Furthermore, as displayed in Table 1 at 9:08, an action at one layer may not correspond to an action in another layer at all. Most interestingly, we discovered that knowledge authoring is full of what we call *Hidden Acts*, which do not correspond with actual Protégé actions, but which turn out to play an important role in the interaction, as evidenced by subjects’ spoken utterances. These include, most prominently, various types of goal handling, such as the adoption, revision, satisfaction, and abandonment of goals. At the dialogue level, these correspond to the expression of desires (when a goal is set aside for later) and intentions (when a goal is starting to be pursued).

How does our annotation scheme compare to schemes for the analysis of natural language dialogue, such as the ISO-standard of (Bunt et al., 2010)? Similarities with our own scheme are easy to find: the distinction between information seeking and informative providing is an obvious example. Important differences exist as well. The ISO standard has been developed for annotating a wide class of dialogues. Our own scheme targets a specific kind of “asymmetric” dialogue, between a person and a machine. Goals can only be expressed by the person; *what-if* questions can only be posed by the person (the system responds, for example, with an *Expressive Statement*). Furthermore, the person cannot, at present, make

Expressive Statements (given that the addition of an axiom has been modelled as a *Command*). *Proposals* (as opposed to *Commands*) are only open to the system. Perhaps the most interesting differences between the two schemes relate to the expression of goals, which appears to be absent from the ISO standard. A few other natural-language related annotation schemes (e.g, (Leech and Weisser, 2003)) do allow the expression of "desires", though the distinction between desires and intentions – reminiscent of Bratman-style "Beliefs, Desires and Intentions" (Bratman, 1999) – is not made.

4 Where do we go from here?

Our annotation scheme has been updated to reflect the observations reported in the previous section, adding categories like *Expression of Intention* and *Expression of Desire*. Having offered tentative answers to the three questions mentioned at the end of the introduction to this paper, how do we plan to proceed?

First, we aim to further improve our understanding of knowledge authoring. Our experiment has given us some relevant insights, as incorporated in our modified annotation scheme, but the validity of these insights has yet to be proven. To do this we shall write an explicit annotation manual, and ask annotators to use these in a full-scale experiment. As before, the study proper will have as participants ontologists working on an ontology they either own or collaborate on in authoring and will not necessarily be computer scientists. Thus the study proper will observe participants in their everyday work, rather than a task that was merely an artefact of the study. In such a setting, there is the potential for a *think aloud* protocol to distort 'normal working practices', which was not an issue in the pilot study. After the experiment they will be asked to clarify their actions (retrospective think-aloud), and to elaborate on what they were trying to do (Olmsted-Hawala et al., 2010). Next, we shall ask annotators to use the annotation manual (and an annotation tool based on the categories defined in the manual) to annotate a number of episodes from the full-scale experiment. Inter-annotator agreement will then be measured using standard methods (for example, the κ test), which should tell us whether our annotation scheme, and the annotations that are based on it, are clear. Where necessary the scheme will be adapted in light of the experiment and the editors' clarifications.

As explained in the Introduction, we aim to exploit our improved understanding of knowledge authoring by designing an improved knowledge authoring interface. This will be a difficult step, which will start with an in-depth study of the outcomes of the full-scale experiment, trying to understand what ontology authors were attempting to do, and how their actions may be better supported by an improved knowledge authoring interface. It is too early to say with certainty what these improvements will be but, in light of our pilot studies, it seems plausible that the new interface will allow ontology authors to manipulate goals at different levels ("I'd like this concept to be subsumed by concept X", "I'd like this concept to have an infinite extension", "This goal has now been fulfilled and can therefore be deleted"), ask competency questions (such as questions that knowledge authors are often asked to write down before starting the authoring process proper) and ask *what-if* questions that allow users to explore the consequences of potential actions without as yet committing to them. Ultimately, we envisage implementing the new dialogue moves as a plug-in to Protégé-style knowledge authoring software.

References

Abgaz, Y., M. Javed, and C. Pahl (2012). Analyzing impacts of change operations in evolving ontologies. In *ISWC Workshops: Joint Workshop on Knowledge Evolution and Ontology Dynamics*.

Bernardi, R., D. Calvanese, and C. Thorne (2012). Designing efficient controlled languages for ontologies. In *Computating Meaning* (Vol 4 ed.). Springer.

Beun, R. (1989). *The recognition of declarative questions in information dialogues*. Ph. D. thesis, University of Tilburg.

Bratman, M. (1999). *Intention, plans, and practical reason*. Cambridge University Press.

Bunt, H. (1994). Context and dialogue control. *Think Quarterly 3*(1), 19–31.

Bunt, H., J. Alexandersson, J. Carletta, J.-W. Choe, A. C. Fang, K. Hasida, K. Lee, V. Petukhova, A. Popescu-Belis, L. Romary, C. Soria, and D. Traum (2010). Towards an iso standard for dialogue act annotation. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*, Valletta, Malta. European Language Resources Association.

Buxton, W. (1983). Lexical and pragmatic considerations of input structures. *Computer Graphics 17*(1), 31–37.

Card, S., T. Moran, and A. Newell (1986). *The psychology of human-computer interaction.* L. Erlbaum Associates Inc.

Dzbor, M., E. Motta, J. M. Gomez, C. Buil, K. Dellschaft, O. Görlitz, and H. Lewen (2006). D4.1.1 analysis of user needs, behaviours & requirements wrt user interfaces for ontology engineering. Technical report, NeOn Project.

Foley, J. and A. Van Dam (1982). *Fundamentals of interactive computer graphics*, Volume 1. Addison Wesley Longman Publishing Co.

Fuchs, N., U. Schwertel, and R. Schwitter (1999). Attempto controlled english not just another logic specification language. In *Logic-Based Program Synthesis and Transformation*, Volume 1559, pp. 1–20. Springer.

Ginzburg, J. (1994). An update semantics for dialogue. In *Proceedings of the 1st International Workshop on Computational Semantics.*

Goncalves, R. S., B. Parsia, and U. Sattler (2011). Categorising logical differences between owl ontologies. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 1541–1546. ACM.

Leech, G. and M. Weisser (2003). *Generic speech act annotation for task-oriented dialogues*, pp. 441–446. Centre for Computer Corpus Research on Language Technical Papers, Lancaster University.

Moran, T. (1981). The command language grammar: A representation for the user interface of interactive computer systems. *International Journal of Man-Machine Studies 15*(1), 3–50.

Nielsen, J. (1986). A virtual protocol model for computer-human interaction. *International Journal of Man-Machine Studies 24*(3), 301–312.

Olmsted-Hawala, E., E. Murphy, S. Hawala, and K. Ashenfelter (2010). Think-aloud protocols: a comparison of three think-aloud protocols for use in testing data-dissemination web sites for usability. In *the 28th international conference on Human factors in computing systems*, pp. 2381–2390.

Power, R. (2009). Towards a generation-based semantic web authoring tool. In *Proceedings of the 12th European Workshop on Natural Language Generation*, Stroudsburg, PA, USA, pp. 9–15. Association for Computational Linguistics.

Rector, A., N. Drummond, M. Horridge, J. Rogers, H. Knublauch, R. Stevens, H. Wang, and C. Wroe (2004). Owl pizzas: Practical experience of teaching owl-dl: Common errors & common patterns. In *Engineering Knowledge in the Age of the Semantic Web*, Volume 3257, pp. 63–81. Springer.

Schwitter, R. (2002). English as a formal specification language. In *Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on*, pp. 228 – 232.

Sowa, J. (2004). Common logic controlled english. `http://www.jfsowa.com/clce/specs.htm`. Accessed: 27/11/2012.