AlvisAE: a collaborative Web text annotation editor for knowledge acquisition

Frédéric Papazian **Robert Bossy** **Claire Nédellec**
Mathématique, Informatique et Génome, Institut National de la Recherche Agronomique
INRA UR1077 – F78352 Jouy-en-Josas
{forename.lastname}@jouy.inra.fr

Abstract

AlvisAE is a text annotation editor aimed at knowledge acquisition projects. An expressive annotation data model allows AlvisAE to support various knowledge acquisition tasks like construction gold standard corpus, ontology population and assisted reading. Collaboration is achieved through a workflow of tasks that emulates common practices (*e.g.* automatic pre-annotation, adjudication). It is implemented as a Web application requiring no installation by the end-user, thus facilitating the participation of domain experts. AlvisAE is used in several knowledge acquisition projects in the domains of biology and crop science.

1 Introduction

Text annotation editors have become key tools in various fields of research like Computational Linguistics, Information Extraction, Text Mining or Semantic Web. The requirements of each specific community drive the implementation of annotation editors developed in the past ten years. We advance AlvisAE, an annotation editor that focuses on semantic annotation for the purpose of knowledge acquisition and formal modeling in specific domains. There are several uses for text annotations in knowledge acquisition among which three are enumerated in the following:

1. Machine Learning-based Information Extraction systems capture the knowledge contained in a domain speech. But they require training sets; annotation editors are essential tools to build gold standards from corpus, but, provided they have the appropriate facilities, they can also assist the design of the annotation guidelines and the supervision of the annotation quality (*e.g.* Inter-Annotator Agreement scores, adjudication features).

2. Annotation editors are powerful companion tools for ontology population and terminology design. Indeed, they allow annotators to access and select domain terms and concepts in their speech context and to establish explicit relationships between the lexical level and the conceptual level. Thus, by providing a user-friendly interface, annotation editors help to choose more relevant terms and concept labels together with their definition and to discover semantic relations between concepts.

3. In the context of Information Retrieval, the Annotation Editor can provide reading assistance by highlighting relevant concepts and relationships within the text. The annotation editor can also empower the users to give feedback about the Information Retrieval results and then about the domain model.

AlvisAE is an annotation editor and framework implemented with these goals. It supports an expressive annotation schema language that allows to specify a wide variety of annotation tasks including: automatic supporting linguistic annotations (*e.g.* tokenization, POS tagging, NER, parsing, anaphora), text-bound annotation (*e.g.* named-entities, terms), semantic relations and events and ontology population. AlvisAE also supports collaborative annotation

Proceedings of the 6th Linguistic Annotation Workshop, pages 149–152,
Jeju, Republic of Korea, 12-13 July 2012.

through the definition of a workflow that specifies a sequence of tasks. By breaking an annotation project into tasks, AlvisAE facilitates the division of work among annotators according to their skills. Finally the AlvisAE client is a full Web application that requires only a modern browser to operate, in this way it targets any domain expert regardless of their workstation device.

In section 2 we discuss related work, then we describe AlvisAE principles and implementation in section 3. Finally, we present ongoing projects using AlvisAE and our plans for the future in section 4.

2 Related work

Semantic annotation of text requires that annotators can express complex bits of knowledge through the editor data model. The benefit of allowing the annotation of relations is attested, although most annotations editors are limited to text span annotations. A major challenge of the annotation of relations is the representation on screen. Indeed, the most natural way to display relations is graphically, by a line between the relation arguments. However lines can disrupt the reading flow if they cross or hide the text and thus can hinder the annotator productivity. Some tools like Glozz (Widlöcher and Mathet, 2009) and BRAT (Stenetorp et al., 2012) have proposed original and non-intrusive displays for relational data, like improved line routing algorithms or a tabular display next to the text.

Collaborative annotation has been a vibrant topic in the recent years because (1) the Web application technologies are becoming mature enough to deal with large collaborative projects, and (2) virtual markets like Amazon's Mechanical Turk raise the expectations of available workforce and offer a new reward scheme for annotators. The most basic collaboration form is the Optimistic Concurrency Control, where concurrent commits are considered to be independent. Knowledge acquisition requires more elaborate collaboration schemes because knowledge models are often the result of a consensus between annotators. A few frameworks go a step beyond by providing a finer control over concurrency as well as a true model of collaboration. For example, GATE Teamwork (Kalina et al., 2010) includes a workflow engine in order to specify the sequence of tasks that will ensure a complete annotation of each document. This work is particularly interesting because the authors advance general types of tasks specific to text annotation projects: automatic annotation tasks by the GATE pipeline, manual annotation tasks and adjudication tasks.

Finally, the most recently developed editors are Web applications like Serengeti (Stührenberg et al., 2007), BRAT (Stenetorp et al., 2012) or ODIN (Rinaldi et al., 2010). As stated above, the libraries for building browser-based clients have reached a level of stability that allows their extensive use. Moreover, Web applications have very low system requirements for the end user thus ensuring a wider community of annotators, in particular domain experts.

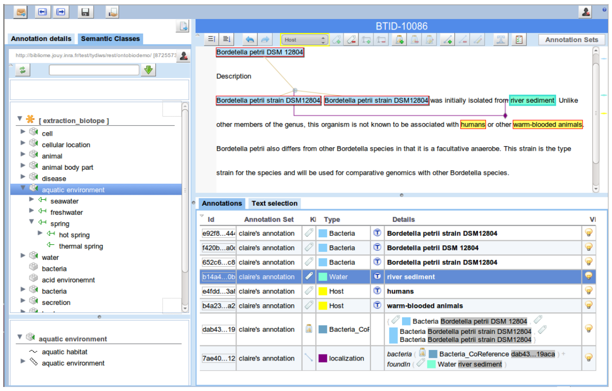
3 Description of AlvisAE

The AlvisAE architecture consists of a RESTful server and a Web application client. The server has the responsibility for the storage of documents and annotations, for authentication and authorization of the annotators, and for workflow enforcement. The client is a Web application that allows the user to log in, to request documents and tasks and to visualize and to edit annotations. Figure 1 illustrates the interaction of the user with AlvisAE.

3.1 Annotation Model

The AlvisAE annotation model has been designed to encompass the requirements of knowledge acquisition projects. An AlvisAE project must specify an annotation schema that enumerates a set of annotation types. These types usually represent operational categories of annotations (*e.g.* named-entity types, relations). The schema also specifies that each type of annotation belongs to one of the three kinds described in the following:

Text-bound annotations are directly linked to the text of the document by their character position. AlvisAE supports enclosing, overlapping and discontinuous text-bound annotations. Discontinuous annotations are bound to a set of fragments of the document text; they allow to represent entities that are spread in different locations of a sentence, such as coordinated modifiers with the same head (*e.g.*



Bordetella petrii DSM 12804

Description

Bordetella petrii strain DSM12804. Bordetella petrii strain DSM12804 was initially isolated from river sediment. Unlike other members of the genus, this organism is not known to be associated with humans or other warm-blooded animals. Bordetella petrii also differs from other Bordetella species in that it is a facultative anaerobe. This strain is the type strain for the species and will be used for comparative genomics with other Bordetella species.

Id	Annotation Set	Ki	Type		Details	Vi
e92f8...44	claire's annotation		Bacteria	Ⓣ	Bordetella petrii strain DSM12804	
f420b...a0	claire's annotation		Bacteria	Ⓣ	Bordetella petrii DSM 12804	
652c6...c8	claire's annotation		Bacteria	Ⓣ	Bordetella petrii strain DSM12804	
b14a4...0b	claire's annotation		Water	Ⓣ	river sediment	
e4fdd...3a	claire's annotation		Host	Ⓣ	humans	
b4a23...a2	claire's annotation		Host	Ⓣ	warm-blooded animals	
dab43...19	claire's annotation		Bacteria_CoF		{ Bacteria Bordetella petrii DSM 12804 , Bacteria Bordetella petrii strain DSM12804 , Bacteria Bordetella petrii strain DSM12804 }	
7ae40...12	claire's annotation		localization		bacteria (Bacteria_CoReference dab43...19aca) + foundIn (Water river sediment)	

Figure 1: **AlvisAE client screen capture.** The upper-right panel displays the text and the annotations: text-bound annotations are highlighted, relations are lines connected with a lozenge, and groups are lines connected with a circle. The lower-right panel is a tabular representation of all annotations in the current document; the user can select and navigate by using either panels. The left panel shows an ontology that is being collaboratively designed; users can drag and drop text-bound annotations to create new concept labels and synonyms.

“North and South America”). A type of text-bound annotations can be constrained to token boundaries.

Relations Relation annotations are tuples of annotations; each argument is labelled with a role. The annotation schema can specify the types of annotations allowed for each role. AlvisAE is not restricted to text-bound relation arguments, meaning that there can be higher-order relations (*e.g.* relations of relations). Even though most relations are binary, AlvisAE supports relations of arbitrary arity. Relations are displayed either in the table layout, or as lines connecting arguments, nevertheless they can be hidden to improve the readability.

Groups Group annotations are collections of annotations; group elements are neither labelled or ordered. Groups are useful to connect an arbitrary number of annotations, for instance to represent coreference chains. In the same way as relations, groups can contain annotations of any kind.

Additionally all annotations have properties in the form of key-value pairs. The schema can express standard constraints on property values (*e.g.* closed value set, numeric range). Furthermore, property values can be bound to an external resource like an ontology or a terminology. In the screen capture (figure 1), the left layout shows a shared termino-ontology managed by the TyDI software (Nedellec et al., 2010). Text-bound annotations can be added as new terms or synonyms in the terminology (left layout) or as new concept labels with a simple simple drag-and-drop operation.

3.2 Annotation Task Workflow

Collaborative annotation with AlvisAE is supported through the definition of a workflow in a similar way as with Teamware (Kalina et al., 2010). The workflow is a set of tasks; each task is an atomic unit of annotation work that covers a subset of annotation types of the schema. Different tasks for the same document can be assigned to different annotators. In this way, the tasks can be dispatched according to the skill of each annotator. For example, junior domain experts can be assigned to the named-entities annotation task, natural language experts can be assigned to the coreference annotation task, and senior domain experts can be assigned to domain-specific relation annotation task. AlvisAE supports pre-annotation by an automatic corpus processing as a task to be assigned to a software agent instead of a human annotator. For example, AlvisAE can easily call the AlvisNLP (Nédellec et al., 2009) corpus processing engine that includes the most common NLP tasks.

AlvisAE workflow also specifies for each task a *cardinality* that is the number of annotators that must perform this task for each document. A cardinality of one means that the task is carried out by a single annotator. A cardinality of two emulates the common practice of double annotation.

Finally, a workflow may specify *review* tasks. A review task is bound to a regular annotation task and covers the same annotation types. The annotator assigned to a review is required to go through the annotations created within the scope of the preceding tasks, and to correct them according to the guidelines. If the preceding task has cardinality greater than one, then the annotator has to review all the concurrent annotations and pull out a consensus. In other words review tasks are adjudication tasks where the cardinality is greater than one.

The order in which tasks are performed on a document is constrained by both the schema and the required reviews. Tasks that cover compound annotations types (relations and groups) depend on the the tasks that cover the annotation types of their arguments and elements. Reviews depend on the tasks to which they are bound by definition. AlvisAE checks the consistency of the workflow against straightforward rules (*e.g.* all annotation types must be covered by a task, circular workflows are invalid, tasks with cardinality greater than one must be reviewed). More importantly, the characterization of the workflow ensures a full traceability of knowledge model produced collectively by the annotators.

4 Applications and Future Work

AlvisAE is currently used in several funded projects in the domains of biology and crop science, although it is not restricted to these domains:

OntoBiotope aims at building an ontology of bacteria habitats and tropisms as well as the annotation of a training corpus for Information Extraction systems.

FSOV SAM gathers knowledge about the relationships between phenotypes, genes and markers in a corpus of wheat genetics literature.

Bacteria Gene Interactions designs training corpus for Information Extraction systems about genic interactions in bacteria. This project is a follow-up of the BioNLP Bacteria Gene Interaction shared task (Bossy et al., 2012).

Our future efforts will concentrate in the development of adjudication tools and interface. The main challenge lies on the simultaneous alignment of several kinds of annotations. Indeed, the adjudication of compound annotations (relations and groups) depends on the prior adjudication of their arguments.

Currently, the specification of a schema and a workflow rely on two configuration files in XML, and the set up of an AlvisAE project is done by a command-line interface. We plan to develop a Web client dedicated to project management including its creation, definition and supervision.

References

Robert Bossy, Julien Jourde, Alain-Pierre Manine, Philippe Veber, Erick Alphonse, Maarten Van De Guchte, Philippe Bessières, and Claire Nédellec. 2012. BioNLP 2011 Shared Task - The Bacteria Track. *BMC Bioinformatics*, 13(suppl. 8):S3.

Bontcheva Kalina, H. Cunningham, I. Roberts, and V. Tablan. 2010. Web-based collaborative corpus annotation: Requirements and a framework implementation. In *New Challenges for NLP Frameworks (LREC)*, Malta, May.

Claire Nedellec, Wiktoria Golik, Sophie Aubin, and Robert Bossy. 2010. Building large lexicalized ontologies from text: A use case in automatic indexing of biotechnology patents. In *EKAW*, pages 514–523.

Claire Nédellec, Adeline Nazarenko, and Robert Bossy. 2009. Information extraction. In Peter Bernus, Jacek Blazewicz, Günter J. Schmidt, Michael J. Shaw, Steffen Staab, and Rudi Studer, editors, *Handbook on Ontologies*, International Handbooks on Information Systems, pages 663–685. Springer, Berlin Heidelberg.

Fabio Rinaldi, Simon Clematide, Gerold Schneider, Martin Romacker, and Thérèse Vachon. 2010. Odin: An advanced interface for the curation of biomedical literature. In *Fourth International Biocuration Conference*.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France, April. Association for Computational Linguistics.

Maik Stührenberg, Daniela Goecke, Nils Diewald, Irene Cramer, and Alexander Mehler. 2007. Web-based annotation of anaphoric relations and lexical chains. In Branimir Boguraev, Nancy M. Ide, Adam Meyers, Shigeko Nariyama, Manfred Stede, Janyce Wiebe, and Graham Wilcock, editors, *Proceedings of the Linguistic Annotation Workshop*, pages 140–147, Prague. Association for Computational Linguistics.

Antoine Widlöcher and Yann Mathet. 2009. La plate-forme glozz : environnement d'annotation et d'exploration de corpus. TALN, Senlis, France.