ISO/TC 37/SC 4 N079

ISO/TC 37/SC 4/WG Nxx

ISO/TC 37/SC 4

2003-02-16

Language Resource Management

Secretariat: Korterm

Linguistic resource management –Linguistic annotation framework

Warning

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard. Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

internet: iso@iso.ch

Introduction

Language resources are bodies of electronic language data used to support research and applications in the area of natural language processing. Typically, such data are enhanced (annotated) with linguistic information such as morpho-syntactic categories, syntactic or discourse structure, co-reference information, etc.; or two or more bodies may be aligned for correspondences (e.g., parallel translations, speech signal and transcription).

Over the past 15-20 years, increasingly large bodies of language resources have been created and annotated by the language engineering community. Certain fundamental representation principles have been widely adopted, such as the use of stand-off annotation, use of XML, etc., and several attempts to provide generalized annotation mechanisms and formats have been developed (e.g., XCES, annotation graphs). However, it remains the case that annotation formats often vary considerably from resource to resource, often to satisfy constraints imposed by particular processing software. The language processing community has recognized that commonality and interoperability are increasingly imperative to enable sharing, merging, and comparison of language resources. Therefore, to provide an infra-structure and framework for language resource development and use, the International Organization for Standardization (ISO) has formed a sub-committee (SC4) under Technical Committee 37 (TC37, Terminology and Other Language Resources) devoted to Language Resource Management. Within this sub-committee, a working group (WG1-1) has been established to develop a Linguistic Annotation Framework that can serve as a basis for harmonizing existing language resources as well as developing new ones.

Recognizing the diverse needs for representing different types of linguistic annotation, divergences in theoretical approach, and the existence of large bodies of legacy data and software, WG1-1 does not seek to establish a single, definitive annotation scheme or format. Rather, the goal is to provide a framework for linguistic annotation of language resources that can serve as a reference or pivot for different annotation schemes, and which will enable their merging and/or comparison. To this end, the work of WG1-1 will include the following:

(1) analysis of the full range of annotation types and existing schemes, to identify the fundamental structural principles and content categories;

(2) instantiation of an abstract format capable of capturing the structure and content of linguistic annotations, based on the analysis in (1);

(3) establishment of a mechanism for formal definition of a set of reference content categories which can be used “off the shelf” or serve as a point of departure for precise definition of new or modified categories.

(4) provision of both a set of guidelines and principles for developing new annotation schemes and concrete mechanisms for their implementation, for those who wish to use them.

To establish a basis for development of a linguistic annotation framework, a workshop of experts was convened on November 21-22, 2002, at Pont-à-Mousson, France. This document summarizes the discussions and conclusions from this workshop, which included the following participants:

BEL, Nuria, Universitat de Barcelona
DURAND, David, Brown University
THOMPSON, Henry, University of Edinburgh

HASIDA, Koiti, AIST Tokyo
DE LA CLERGERIE, Eric, INRIA
CLEMENT, Lionel, INRIA
ROMARY, Laurent, LORIA
IDE, Nancy, Vassar College
LEE, Kiyong, Korea University
SUDERMAN, Keith, Vassar College
KUMAR, Aswani, LORIA
LAPRUN, Chris, NIST
DECLERCK, Thierry, DFKI
CARLETTA, Jean, University of Edinburgh
STRUBE, Michael, European Media Laboratory
CUNNINGHAM, Hamish, University of Sheffield
ERJAVEC, Tomaz, Institute Jozef Stefan
BRUGMAN, Hennie, Max-Planck-Institut für Psycholinguistik
VITALI, Fabio, Universite di Bologna
CHOI, Key-Sun, Korterm
BORDE, Jean-Michel, Digital Visual
KOW, Eric, LORIA

General requirements for a linguistic annotation framework

The following general requirements for a linguistic annotation framework were identified:

Expressive adequacy

- The framework must provide means to represent all varieties of linguistic information (and possibly also other types of information). This includes representing the full range of information from the very general to information at the finest level of granularity.

Media independence

- The framework must handle all potential media types, including text, audio, video, image, etc. and should, in principle, provide common mechanisms for handling all of them. The framework will rely on existing or developing standards for representing multi-media.

Semantic adequacy

- Representation structures must have a formal semantics, including definitions of logical operations
- There must exist a centralized way of sharing descriptors and information categories

Incrementality

- The framework must provide support for various stages of input interpretation and output generation.
- The framework must provide for the representation of partial/under-specified results and ambiguities, alternatives, etc. and their merging and comparison.

Uniformity

- Representations must utilize same “building blocks” and the same methods for combining them.

Openness

- The framework must not dictate representations dependent on a single linguistic theory.

Extensibility

- The framework must provide ways to declare and interchange extensions to the centralized data category registry.

Human readability

- Representations must be human readable, at least for creation and editing.

Processability (explicitness)

- Information in an annotation scheme must be explicit—that is, the burden of interpretation should not be left to the processing software.

Consistency

- Different mechanisms should not be used to indicate the same type of information.

To fulfill these requirements, it is necessary to identify a consistent underlying *data model* for data and its annotations. A data model is a formalized description of the data objects (in terms of composition, attributes, class membership, applicable procedures, etc.) and relations among them, independent of their instantiation in any particular form. A data model capable of capturing the structure and relations in diverse types of data and annotations is a pre-requisite for developing a common corpus-handling environment: it impacts the design of annotation schema, encoding formats and data architectures, and tool architectures.

As a starting assumption, we can conceive of an annotation as a one- or two-way link between an annotation object and a point (or a list/set of points) or span (or a list/set of spans) within a base data set. Links may or may not have a semantics--i.e., a type--associated with them. Points and spans in the base data may themselves be objects, or sets or lists of objects. We make several observations concerning this assumption:

- the model assumes a fundamental linearity of objects in the base,[1] e.g., as a time line (speech); a sequence of characters, words, sentences, etc.; or pixel data representing images;
- the *granularity* of the data representation and encoding is critical: it must be possible to uniquely point to the smallest possible component (e.g., character, phonetic component, pitch signal, morpheme, word, etc.);
- an annotation scheme must be mappable to the structures defined for annotation objects in the model;
- an encoding scheme must be able to capture the object structure and relations expressed in the model, including class membership and inheritance, therefore requiring a sophisticated means to specify linkage within and between documents;
- it is necessary to consider the logistics of identifying spans by enclosing them in start and end tags (thus enabling hierarchical grouping of objects in the data itself), vs. explicit addressing of start and end points, which is required for read-only data;

[1] Note that this observation applies to the *fundamental* structure of stored data. Because the targets of a relation may be either individual objects, or sets or lists of objects, information with more than one dimension is accommodated.

- it must be possible to represent objects and relations in some (fairly straightforward) form that prevents information loss;
- ideally, it should be possible to represent the objects and relations in a variety of formats suitable to different tools and applications.

The framework for linguistic annotation should allow for variation in annotation schemes while at the same time enabling comparison and evaluation, merging of different annotations, and development of common tools for creating and using annotated data. For this purpose we envisage a common "pivot" format based on a data model capable of capturing all types of information in linguistic annotations, into and out of which site-specific representation formats can be transduced.

Terms and definitions

Annotation

In this document the term *annotation* refers to the process of adding linguistic information to language data ("annotation of a corpus") or the linguistic information itself ("an annotation"), independent of its representation. For example, one may annotate a document for syntax using a LISP-like representation, an XML representation, etc.

Representation

The term *representation* refers to the format in which the annotation is rendered, e.g. XML, LISP, etc. independent of its content. For example, a phrase structure syntactic annotation and a dependency-based annotation may both be represented using XML, even though the annotation information itself is very different.

Types of Annotation

We distinguish two fundamental types of annotation activity:

1. *segmentation* : delimits linguistic elements that appear in the primary data. including
    a. continuous segments (appear contiguously in the primary data)
    b. super- and sub-segments, where groups of segments will comprise the parts of a larger segment (e.g., a contiguous word segments typically comprise a sentence segment)
    c. discontinuous segments (linking continuous segments)
    d. landmarks (e.g time stamps) that note a point in the primary data

    In current practice, segmental information may or may not appear in the document containing the primary data itself. Documents considered to be *read-only*, for example, might be segmented by specifying byte offsets into the primary document where a given segment begins and ends.
2. *linguistic annotation* : provides linguistic information about the segments in the primary data, e.g., a morpho-syntactic annotation in which a part of speech and lemma are associated with each segment in the data. Note that the identification of a segment as a word, sentence, noun phrase, etc. also constitutes linguistic annotation. In current practice, when it is possible to do so, segmentation and identification of the linguistic role or properties of that segment are often combined (e.g., syntactic bracketing, or

delimiting each word in the document with an XML tag that identifies the segment as a word, sentence, etc.).

Stand-off annotation

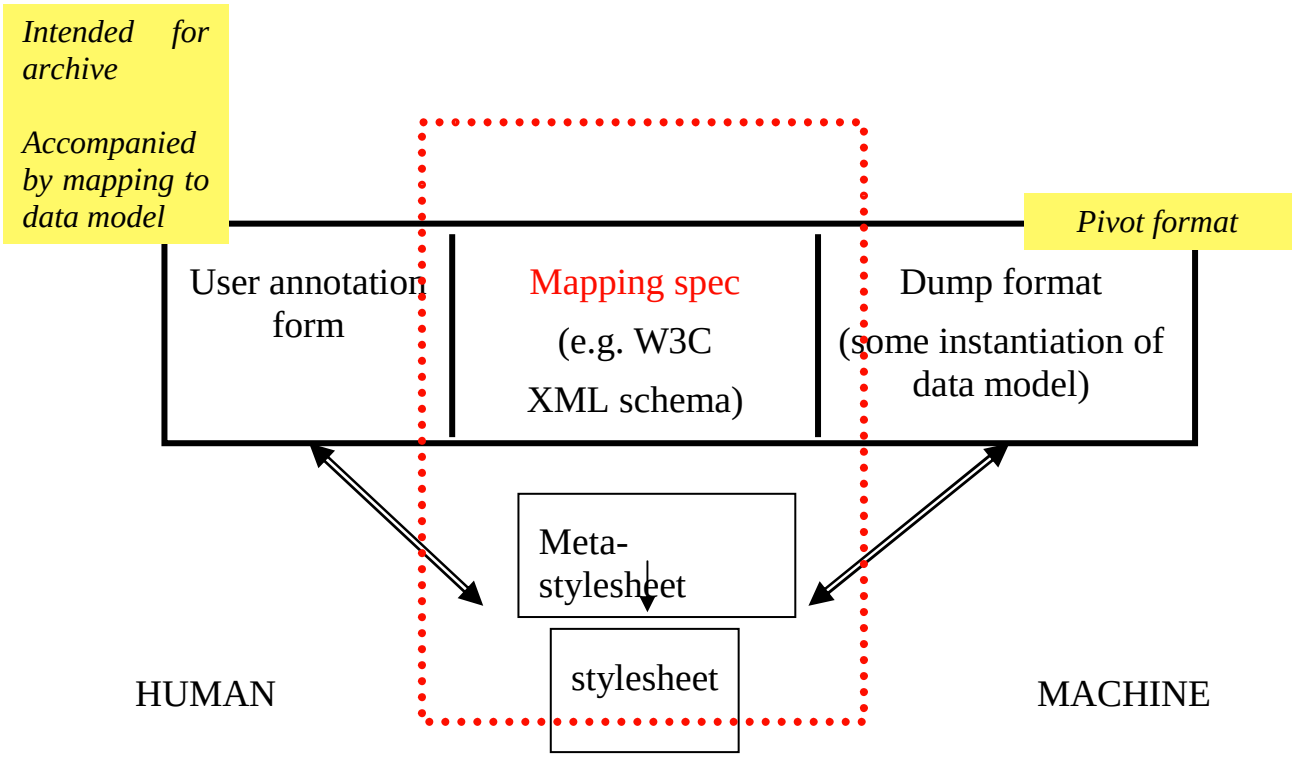
Annotations layered over a given primary document and instantiated in a document separate from that containing the primary data. Stand-off annotations refer to specific locations in the primary data, by addressing byte offsets, elements, etc. to which the annotation applies. Multiple stand-off annotation documents for a given type of annotation can refer to the same primary document (e.g., two different part of speech annotations for a given text). There is no requirement that a single XML-compliant document may be created by merging stand-off annotation documents with the primary data; that is, two annotation documents may specify trees over the primary data that contain overlapping hierarchies.

Design principles

The workshop participants identified the following general principles to guide the development of the linguistic annotation framework:

- The data model and document form are distinct but mappable to one another
- The data model is parsimonious, general, and formally precise
- The data model is built around a clear separation of structure and content
- There is an inventory of logical operations supported by the data model, which define its abstract semantics
- The document form is largely under user control
- The mapping between the flexible document form and data model is via a rigid dump-format
- The mapping from document form to the dump format is documented in an XML Schema (or the functional equivalent thereof) associated with the document
- Mapping is operationalized *either* via schema-based data-binding process *or* via schema-derived stylesheet mapping between the user document and the dump-format document.
- It must be possible to isolate specific layers of annotation from other annotation layers or the primary (base) data; i.e., it must be possible to create a format using stand-off annotation
- The dump format must be designed to enable stream marshalling and unmarshalling

Based on these principles, we envisaged the overall structure of the linguistic annotation framework as shown below:



Or the functional equivalent thereof

The left side of the diagram represents the user-defined document form, and is labeled "human" to indicate that creation and editing, of the resource is accomplished via human interaction with this format. This format should, to the extent possible, be human readable. We will support XML for these formats (e.g., by providing style sheets, examples, etc.) but not disallow other formats.

The right side represents the dump format, which is machine processable, and may not be human readable as it is intended for use only in processing. This format will be instantiated in XML.

Practice

Once the overall shape of the linguistic annotation framework was identified, the participants specified the following set of practices for its implementation:

- The data model is essentially a feature structure graph with a moderate admixture of algebra (e.g. disjunction, sets), grounded in *n*-dimensional regions of primary data and literals.
- The dump format is isomorphic to data model.
- Semantic coherence is provided by a registry of features in an XML-compatible format (e.g., RDF), which can be used directly in the user-defined formats and is always used with the dump format.
- Resources will be available to support the design and specification of document forms, for example:

    - XML Schemas in several normal forms based on type definitions and abstract elements which can be exploited via type derivation and/or substitution group;
    - XPointer design-patterns with standoff semantics;
    - Schema annotations specifying mapping between document form and data model;
    - Meta-stylesheet for mapping from annotated XML Schema to mapping stylesheets;
    - Data-binding stylesheets with language-specific bindings (e.g. Java).
- Users may define their own data categories or establish variants of categories in the registry. In such cases, the newly defined data categories will be formalized using the same format as definitions available in the registry, and will be associated with the dump format.
- The responsibility of converting to the dump format is on the producer of the resource.
- The producer is responsible for documenting the mapping from the user format to the data model
- The ISO working group will provide test suites and examples following these guidelines:
    - The example format should illustrate use of data model/mapping
    - The examples will show both the left and right side formats
    - Examples will be provided that use existing schemes

Work to be completed for Sapporo Meeting (July 2003, ACL03):

1. Infrastructure, Editor: Nancy Ide (NWI)
    a. Data model (HT, FV, DD, HC, CL, JC, HB)
        - Ontology + logic
    b. Registry
    c. Dump format (serialization)
2. General terminology, Editor: Key-Sun Choi
3. Morpho-syntax (testbed), Editor: People in France (NWI)
4. Feature structures (example AML), Editor: Kiyong Lee (NWI)