# **Progress on Multi-lingual Named Entity**

# Annotation Guidelines using RDF(S)

# Nigel Collier\*, Koichi Takeuchi\*, Chikashi Nobata<sup>†</sup>, Junichi Fukumoto<sup>‡</sup>, Norihiro Ogata<sup>?</sup>

\*National Institute of Informatics 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430, Japan {collier, koichi}@nii.ac.jp

<sup>†</sup>Communications Research Laboratory 2-2-2 Hikaraidai, Seika-cho, Soraku-gun, Kyoto, Japan

nova@crl.go.jp

<sup>‡</sup>Ritsumeikan University Noji-higashi, Kusatsu-shi, Shiga 525-8577, Japan fukumoto@cs.ritsumei.ac.ip

<sup>?</sup> Osaka University 1-8 Machikaneyama, Toyonaka, Osaka, Japan ogata@lang.osaka-u.ac.jp

#### Abstract

This paper provides a discussion and concise summary of the PIA (Portable Information Access project) guidelines for annotators and tool developers for annotating what we call *named entity 'plus'* (NE+) expressions such as individual names or technical terms that we want to distinguish for whatever reason from the rest of a text. In particular we consider how to annotate locally ambiguous syntactic and semantic structures. We provide notation that conforms to RDF(S) so that annotated documents can have their content accessed on the Semantic Web, i.e. the next generation World Wide Web. In this new framework named entities become instances of concepts in an explicit ontology, and the base text provides links to the annotation and ontology data files.

#### 1. Introduction

The Portable Information Access (PIA) (Collier *et al.* 2001) project aims to develop a domain and language portable information extraction (IE) system. In contrast to other Web-based technologies such as information retrieval (IR) which are characterized by strong portability, no such system as yet exists for IE. PIA is building its foundations on four resources: the PIA annotation tool (PAT), the PIA annotation management system (PAM), the PIA-Core IE module (Collier *et al.* 2002), and PIA annotation guidelines. All of these will be integrated within an online knowledge development environment called 'ontology forge'.

In PIA we are concerned with machine learning for text-to-knowledge conversion so that computer programs can both extract knowledge and also help users to interact intelligently with facts contained in documents. This necessarily involves automatically finding instances of classes and axiomatic relations that are defined in an ontology. This relates to the *semantics* or 'meaning' of the document. We have decided to approach this using supervised machine learning from annotated texts.

PIA guidelines are being drawn up to promote high quality and consistent annotation that will allow instantiation of concepts of interest in a wide range of domains and a linkage back to the mention of the concept in the text. In particular we provide here a concise summary of what we call the named entity plus (NE+) annotation guidelines for English and Japanese and highlight some key issues. The guidelines are based on those developed for the Message Understanding Conference (MUC)-7 task (Chinchor, 1997) and also motivated by existing standardization efforts for text encoding such as the TIPSTER Text Architecture Design (Grishman, 1998) and the Corpus Encoding Standards of EAGLES (Expert Advisory Group on Language Engineering Standards) (Ide & Priest-Dorman, 2000).

In an extension to previous schemes PIA considers NEs to be instances of concepts that are given in an explicit ontology that defines their relations with other concepts and any attribute values that they can take. In other words we include types as well as individuals in our definition of NE+. The annotation of NE+ strongly conforms to the Resource Description Framework Schema 'RDF(S)' notation (Brickley and Guha, 2000) used for making knowledge available to intelligent computer programs on the Semantic Web.

Finally we should say that all PIA guidelines are being developed and refined bottom up in a spirit of cooperation with other interested groups. We therefore actively encourage and welcome feedback from other groups and comments from the research community in general. An initial publication of the full guidelines (v1.4.2) will be released as an NII technical report in the summer of 2002 followed by periodic updates which we intend will include support for Thai and Arabic.

#### 1.1. Relation with MUC-7 NE guidelines

Before proceeding we would like to take care to note that our aim is not to provide criticism of MUCs but to build on the important work that they started. For our purposes in PIA we felt that the MUC-7 guidelines required revision and extension for three main reasons which we will now outline.

Firstly we want to allow markup of terminological expressions which have internal structure (both syntactic and semantic) that sometimes requires explicit disambiguation<sup>1</sup>. The original guidelines focused mainly on names of people, organizations etc. and seemed to provide arbitrary 'fixes' for this issue which we felt could be improved.

Secondly, due to developments in semantic content annotation motivated by the Semantic Web (Berners-Lee et al., 1999) we want to develop an annotation scheme that will conform to this model and link instances of concepts as they appear in texts to a concept taxonomy (part of an ontology)<sup>2</sup>. Since one of our goals in PIA is to help users access textual information in collections that are online we use the Semantic Web model to represent the ontology through the RDF(S). NEs can now be considered to be instances of concepts encoded in RDF. This second reason is perhaps the most important as it allows concept classes (and hence their instances) to have explicit relations and properties and to be reasoned with using axioms. Also by separating the annotation data from the base document we ensure that the software that creates annotations does not edit the base document at all - thus ensuring its integrity.

The third reason is that we felt that it was necessary to make a clear distinction between *what* should be annotated and *how* it should be annotated. In other words we want to encourage annotators to think about the content (the 'what') of their documents rather than the surface forms of names and technical terms. We aim to make guidelines that will focus mainly on helping annotators to map surface forms of whatever terms they are interested in to what we call 'conventional forms' (the 'how'). Since we can not know in advance what classes of information will be needed in a particular domain we do not attempt to make guidelines for this. Hopefully this will be made clearer in Section 2.

In PIA we extend the meaning of NE to include terminological expressions that are recognized and agreed within a community of domain users. We follow the broad definition of a term made in (de Besse et al. 1997) as "A lexical unit consisting of one or more words which represents a concept in a domain". We have decided to keep the label 'NE' because of its familiarity to the research community. Candidates for inclusion as NE+ expressions are:

- proper nouns, e.g. names of people, places
- temporal expressions, e.g. days of the week, dates
- quantity definitions, e.g. names of monetary values, names of stock market indices
- terminological expressions
- certain expressions that share identity with (1-4).

A taggable NE+ must belong to one of the categories of expression mentioned above and belong to a class that appears in the domain ontology.

In PIA we also reconsider the roles of named entity and coreference expressions that were given in MUC-7. In MUC-7 the two tasks were considered somewhat independently – their results were assumed to be merged in the template element task, a higher level IE task. In PIA we merge the two tasks at a much lower level and define *First class (type-1) named entities* as those for which the meaning can be inferred from the surface form alone and *Second class (type 2) named entities* such as anaphoric expressions which depend on a type 1 named entity for their meaning.

Why should we annotate terminological expressions? Basically within domains there is a strong tendency to lexicalize certain expressions such as technical terms and to re-use these expressions in the creation of new technical terms. The vocabulary in a domain is therefore consistent and predictable to some extent. Our hypothesis is that given sufficient examples of terms which have had their meaning explicitly annotated by domain experts, that computers can be trained to recognize the meaning of unseen terms in the same domain and so partially understand the meaning of the text.

As in MUC-7, systems which perform the NE+ task must produce a single unambiguous output that is as close as possible to that produced by a human expert in the domain. There are two issues to consider: the first is the identification of the boundaries of the NE+ and the second is the classification of the text inside the boundaries according to the concept classes available in the ontology. In these guidelines, unlike MUC-7, we do not prescribe to any particular set of concept classes. These must be chosen by consensus among the community of domain users. Instead we concentrate on graphical and syntactic issues of expressions where the boundaries of an NE+ are ambiguous or its basic form (which we call a *conventional form*) is different to the surface form. We hope that by presenting the guidelines in this way that they will be applicable to many domains.

The current guidelines provide support for English and Japanese. Partial support is expected to follow soon for Thai and Arabic. For users of other languages it is hoped that the underlying principals that we provide here, i.e. ontology, conventional forms, RDF(S), will enable these guidelines to be extended to other languages. In all cases we are interested in extending and revising the guidelines to meet the requirements of the user community and welcome comments and feedback. The updates will be made available in periodic revisions and will be issued as NII reports and made available online.

<sup>&</sup>lt;sup>1</sup> We should note though that while the PIA guidelines *allow* users to annotate nested structures they do not *require* them to do so, so a 'flat' style of annotation is also permissible if that is what the user feels is required.

<sup>&</sup>lt;sup>2</sup> It should be noted that our scheme allows multiple ontologies and annotation files to be created for a single base document and indeed for many documents this may be quite natural due to various user groups requiring different levels of detail in annotation. This ensures data integrity as well as a type of 'semantic modularization'. For example consider a document describing the discovery of a new planet - a news domain may only want to know about the name of the planet and its discover, whereas an astronomy domain may want to know specific details about its primary star, its orbit and chemical composition.

# **1.2.** Relation with Corpus Encoding Standard (CES) guidelines

EAGLES/Vassar/CNRS consortium CES The guidelines (Ide & Priest-Dorman, 2000) offer an extensive treatment of annotation for computer processing of text covering grammatical, rendition, sub-sentential, text layout, parallel text alignment, etc. This information is consistent with international standards organization (ISO) codes for encoding and denoting character sets, languages, dates and times etc. and the Text Encoding Initiative (TEI) P4 guidelines. In PIA we are primarily concerned with the semantics of the text, starting with named entities and coreference element and extending to simple facts that are expressed as relations between such elements within a sentence or possibly paragraph. This allows us to focus only on a sub-set of the issues covered in the CES and TEI guidelines.

The CES guidelines are based on annotation using SGML (Standard Generalized Markup Language)<sup>3</sup> and note the problem of nested tags and possibly overlapping elements which occur when we want to describe complex structures in the text. Like the TEI guidelines, we consider that annotators will wish to markup nested rather than flat term structure (although our scheme can be used for both). In the scheme we describe below this is accomplished by using a 'constituents' property in a manner similar to the TIPSTER recommendation (Grishman, 1998) for hierarchical ordering of text spans.

CES concluded that annotations and the original text (the 'base' document) must be separated for clarity as annotations should be processed by suitable interface software to make them presentable for human understanding. This is something that we fully support and which is natural in the Semantic Web model and its support software that are now emerging. There are several practical reasons for separating base text and annotations as noted in the CES guide part 5, including:

- it avoids the creation of potentially unwieldy documents.
- the original (base) document remains stable and is not modified by any process which may add (or delete) annotation.
- it avoids problems with markup containing overlapping hierarchies (which are not allowed in SGML).
- We can also add,
- it allows annotations to be contributed independently by people other than the creator of the base document, and
- it allows linkage to public ontologies that are created by independent groups to the creator of the base document.

In CES the base document will provide a link to the annotation document. In contrast in PIA annotations themselves encode pointers to the ontology and the base document and all of these pointers are in the forms of URIs. In this respect documents, annotations and ontologies become first class entities that can be referred to and distributed on the Web.

An important issue for us is the mechanism for linking annotations to the base text (called 'locators' in the CES).

The CES guidelines adopt a complex linking mechanism that is based on byte counts in the base document combined with identifier markers based on TEI and HyTime methods. In PIA while we do not require sophisticated linking of annotations to multiple documents we nevertheless felt the need for an expressive, flexible and robust linkage scheme within a single document. Since we are using RDF (Lassila and Swick, 1999) which itself is based on XML syntax, the natural choice of linkage scheme for us is XPointer (DeRose *et al.* 2001) which is a recommendation of the W3C. One advantage for us is that XPointer provides some robustness against changes in the base document.

Unlike the CES (or TEI) guidelines we do not explicitly differentiate between names and technical terms – they are all NEs and instances of classes that are defined in the ontology. It is the role of the ontology to explicitly describe the relation between classes, class attributes and their axiomatic relations. Therefore tag elements such as 'date', 'measure', 'name', 'term' are all subsumed within the 'NAME' element described in these guidelines. Coreferences between elements which share identity such as abbreviations ('abbr' in CES) and their full forms are captured by using the *identity\_id* property and the annotation labels.

# 2. Guidelines

# 2.1. Markup description

The guidelines specify *how to annotate* but not *what to annotate*. The two issues should be considered to be independent to some extent since the first is concerned with a consistent scheme for knowledge markup and the second with ontology creation for which we do not have specific *a priori* knowledge. As was mentioned earlier, the reason for doing this is so that annotators can focus on the content of their documents rather than the surface forms.

From this point we assume that some ontology has been declared and that concept classes have been decided. The types of information that we require for each annotation (instance of a concept) is given in an RDF Schema whose namespace is held on the PIA Web site. The schema is motivated by the annotation scheme used within the Annotea project (Kahan *et al.* 2001) extended to allow for the notion of coreference chains, conventional forms and verification. The idea is that an annotation will be a super-concept from which all other annotatable concepts in the ontology inherit these properties. The name space is described in Table 1.

Property	Description
context	Relates an Annotation to the resource to which the Annotation applies. This is a URI for the base document and an XPointer that specifies the position in the base document of the Annotation.
conventional _form	The conventional form of the annotation (if applicable) as described in the PIA annotation guidelines.
identity_id	A label used for creating coreference chains where the Annotations have identity of reference.

<sup>&</sup>lt;sup>3</sup> Soon to be updated to XML.

orphan	This property takes only Boolean
	values corresponding to 'yes' and 'no'.
	After the annotation is created, if it is
	later detected that the annotation can
	no longer be linked to its correct
	position in doc_id, then this value will
	be set to 'yes' indicating that the
	linkage (in context) needs correcting.
author	The name of the person, software or
	organization most responsible for
	creating the Annotation.
constituents	A list of Annotation labels separated
	by commas representing the immediate
	constituents of this Annotation.
created	The date and time on which the
	Annotation was created. yyyy-mm-
	ddThh:mm:ssZ format recommended.
modified	The date and time on which the
	Annotation was modified. yyyy-mm-
	ddThh:mm:ssZ format recommended.
sure	This property takes only Boolean
	values corresponding to 'yes I am sure'
	and 'no I am not sure' about the
	assignment of this annotation. Used
	primarily in post-annotation
	processing.
comment	A comment that the annotator wishes
	to add to this annotation, possibly used
	to explain an unsure annotation.

 Table 1: The PIA annotation name space: properties of the annotation class

Attributes *author, created, modified* and *comment* all take their definitions from Dublic Core elements (Dublin Core, 1999).

The following simple example should hopefully make this clear. Given a text (1),

Ex.1. In late July, Downing Street announced that Tony Blair and his family were planning to use Easyjet to travel to the south-west of France - the first time the prime minister had traveled on a low-cost airline.

and a simple ontology described in RDF(S) in the namespace 'simple\_ont1' (not shown) that specifies the classes *PERSON*, *ORGANIZATION* and *PLACE*, we can annotate an instance of the PERSON concept 'Tony Blair' as given in example (2).

```
Ex.2.<simple_ont1:PERSON
rdf:about="& simple_ont1;simple_ont1_00021"
<simple_ont1:context>http://www.news.com/page.htm
l#xpointer(string-range(//main,"Tony
Blair")[1])</simple_ont1:context>
<simple_ont1:conventional_form>Tony
Blair</simple_ont1:conventional_form>
<simple_ont1:identity_id>&simple_ont1;00021</sim
ple_ont1:identity>
<simple_ont1:orphan>No</simple_ont1:orphan>
<simple_ont1:author>John</simple_ont1:author>
```

<simple\_ont1:created>2002-03-05T10:36:45Z</simple\_ont1:created> <simple\_ont1:modified>2002-03-05T10:36:45Z</simple\_ont1:modified> <simple\_ont1:sure>Yes</simple\_ont1:sure> <simple\_ont1:comment>Seems okay to me</simple\_ont1:comment> <rdfs:label>00021</rdfs:label> </simple\_ont1:PERSON>

The example illustrates a number of important points. The first is that the instance shown in (2) is declared in the opening element to be of class PERSON in the namespace *simple\_ont1*. Secondly, each instance receives a unique identification label, in this case 00021 which we use, as shown below, to build coreference chains. Thirdly is the use of XPointer to show where in the document the annotation should apply. In this case we have simple said in *xpointer(string-range(//main, "Tony Blair")[1]* that the annotation applies to the first string called "Tony Blair" in the text. It should be noted that XPointer offers a number of different options to allow us to specify the location annotations but for simplicity we have chosen this one.

As mentioned above, the *identity\_id* property is a pointer to the label value of another Annotation and is used to specify equivalence sets of NEs. If co-reference is not being used then *identity\_id* should take the same value as the instances' label. To make this clearer (3) shows where *identity\_id* allows us to specify that 'the prime minister' is a coreference expression dependent on 'Tony Blair' which was specified in instance 00021. In all other respects this coreference expression looks like a normal NE except that it's *conventional\_form* is inherited from the coreferring NE.

Ex.3.<simple\_ont1:PERSON rdf:about="& simple\_ont1;00022" <simple\_ont1:context>http://www.news.com/page.htm *l#xpointer(string-range(//main," the prime minister")[1])</simple\_ont1:context> <simple ont1:conventional form>Tonv* Blair</simple\_ont1:conventional\_form> <simple\_ont1:identity\_id>&simple\_ont1;00021</sim ple\_ont1:identity> <simple\_ont1:orphan>No</simple\_ont1:orphan> <simple\_ont1:author>John</simple\_ont1:author> <simple\_ont1:created>2002-03-05T10:38:45Z</simple\_ont1:created> <simple\_ont1:modified>2002-03-05T10:38:45Z</simple\_ont1:modified> <simple\_ont1:sure>Yes</simple\_ont1:sure> <simple\_ont1:comment>Seems okay to me</simple\_ont1:comment> <rdfs:label>00022</rdfs:label> </simple\_ont1:PERSON>

The *author* property is a string of characters and numbers that indicates the source of the instance. This could be the name of the annotator or computer program, or could be combined with a version or date number. The primary purpose of the source attribute is to distinguish information about the same instance from different annotation sources or for different versions of a source.

Hierarchical ordering of overlapping regions is captured within our scheme by the use of the *constituents* property. It is envisaged that this will work in a similar way to constituency links between spans in the TIPSTER Text Architecture Design recommendations (Grishman, 1998). While the *constituents* property potentially has several applications such as showing syntactic structures, the interpretation as applied to NEs is generally taken to mean a semantic dependency such as the nesting of semantic tags.

The *sure* property is used by the annotator to indicate instances where they are unsure of the annotation: this is used for quality control. If *sure* is set to false then it indicates that the annotator is unsure about the correctness of the annotation and it should be independently checked by another expert. An explanation of why the annotator was unsure will be found *comments*. Otherwise *sure* is set to true.

It is important to note that the above attributes are the minimum set of attributes needed for locating the occurrence of a named entity in a document and are automatically inherited as attributes of all ontology classes in the annotation tool we described in Section 4. The user will of course want to create other attributes for classes depending on his/her needs and the domain.

Finally, to prevent potential confusion and inconsistencies and to avoid redundancy we do not allow instances to be declared which describe the same part of the text with the same class in the ontology.

#### 2.1.1. Description of How to Annotate

The full description of how to annotate is given in the PIA Annotation Guidelines to be released as a technical report in 2002. These will be available from the PIA Web site. Due to space limitations it is not possible here to give more than an idea of the guidelines so we provide examples for English and Japanese in some important areas to illustrate the direction of our work. The guidelines cover five main areas: graphical variations, inflectional variations, shallow syntactic variations such as conjunction, semantic variations including issues of granularity according to the ontology, and discourse variations such as the use of abbreviations, aliases, pronouns and definite descriptions. Some examples are given below to illustrate these issues.

The basic method of how to annotate revolves around the notion of a 'conventional form' of an NE+. In other words we make a distinction between the surface forms and 'conventional form' of a term or expression. Due to the variety of ways in which a term can be expressed, for example resulting from graphical, morphological or shallow syntactic transformations. We consider the 'real' NE to be one of the enumerated set of synonymous NE+s. The one that we choose to call the conventional form may not actually appear in the text itself, but is what most experts in the domain would recognize as something like the basic form of the NE. Hopefully an example should make this clear.

Ex 4. Mr and Mrs Smith

In (4) we consider that there are potentially 3 NE+s which could be found: 'Mr Smith', 'Mrs Smith' and 'Mr and Mrs Smith'. However in order to avoid a potential explosion of terms resulting from conjunctions we do not recommend that 'Mr and Mrs Smith' should be considered as a separate NE+, so in fact these guidelines only recommend annotating two NE+s here. We do not know at this rather shallow level of information extraction which of these entities will be important to the discourse, but there is certainly the potential for the author to provide useful information about either of these entities, so we prefer to annotate each one separately. For example, 'Mr and Mrs Smith arrived at the airshow. Mrs Smith performed the opening ceremony'; 'Mr and Mrs Smith, Chairman and President respectively of XYZ corporation'. In both of these cases we may require knowledge of a different entity.

Conventional forms must be derived directly from the surface text and in general should not include anything that is not present in the surface text. Therefore abbreviations for example are their own conventional forms (and not the full form from which the abbreviation is taken - forming this reference link done through the *identity\_id* property).

Example (5) shows a typical case of shallow syntactic transformation of the conventional form where there is elision of the head in a conjoined expression so that the full meaning can be considered to be 'c-rel (proto) oncogenes' and 'v-rel (proto) oncogenes'.

#### Ex. 5. the c-rel and v-rel (proto) oncogenes

Using the conventional form notation this should be annotated in the molecular biology domain as shown in (6) and (7) in order to clarify the NEs that result from the structural dependency of both expressions on the shared head "(proto) oncogene". Note that both NEs have dropped the plural form "oncogenes". This method of annotation can be extended to arbitrarily long lists of conjoined expressions.

Ex.6.<biol:PROTEIN rdf:about="&biol;0001" <biol:context>http://www.biojournal.com/page.html# xpointer(string-range(//main,"the \_c-rel and v-rel (proto) oncogenes ")[1])</biol:context> <biol:conventional\_form> the c-rel (proto) oncogene </biol:conventional\_form>

#### </biol:PROTEIN>

Ex.7.<bio1:PROTEIN rdf:about="&bio1;0002" <bio1:context>http://www.biojournal.com/page.html# xpointer(string-range(//main,"the \_c-rel and v-rel (proto) oncogenes ")[1])</bio1:context> <bio1:conventional\_form> the v-rel (proto) oncogene </bio1:conventional\_form>

</biol:PROTEIN>

The use of conventional forms also allows us to naturally separate non-intrinsic parts of expressions such as the embedded abbreviation in "The interleukin-2 (IL-2) promoter" in (8) with annotations shown in (9) and (10). Ex.8. The interleukin-2 (IL-2) promoter consists of several independent T cell receptor (TcR) responsive elements.

Ex.9.<bio1:PROTEIN rdf:about="&bio1;00020" <bio1:context>http://www.biojournal.com/page.html# xpointer(string-range(//main,"interleukin-2 (IL-2) promoter")[1])</bio1:context> <bio1:conventional\_form> interleukin-2 promoter </bio1:conventional\_form>

</biol:PROTEIN>

Ex.10.<bio1:PROTEIN rdf:about="&bio1;00021" <bio1:context>http://www.biojournal.com/page.html# xpointer(string-range(//main,"interleukin-2 (IL-2) promoter")[1])</bio1:context> <bio1:conventional\_form>IL-2 </bio1:conventional\_form>

</biol:PROTEIN>

Although there are many principals of conventional forms that can readily be extended to different languages there are inevitably language specific issues which must be considered individually within the framework of the guidelines. For example, one of the most challenging aspects of annotation is for Japanese Kanji compounds. In some respects they raise similar problems to the 'Mr and Mrs Smith' example above. In (11) we want to recover the conventional form of the phrase "衆参両院" (The Upper and Lower House [of the Diet]). In the guidelines we recommend that two of the three possible instances are recorded: '衆院'(The Lower House), '参院'(The Upper House), but not '衆参両院'. This is shown in (12) and (13).

Ex 11. 会社党の一富士鷹夫・新茄子尾連合会長ら が発足させた「民需ドリブル新党準備会」に、会 社党から十人以上の<u>衆参両院</u>議員が参加すること が\*七日固まった。

Ex.12.<simple\_ont1:ORGANIZATION rdf:about="&simple\_ont1;00025" <simple\_ont1:context>http://www.news.co.jp/page.ht ml#xpointer(string-range(//main,"衆参両院 ")[1])</simple\_ont1:context> <simple\_ont1:conventional\_form>衆参 </simple\_ont1:conventional\_form>

</simple\_ont1:ORGANIZATION>

Ex.13.<simple\_ont1:ORGANIZATION rdf:about="&simple\_ont1;00026" <simple\_ont1:context>http://www.news.co.jp/page.ht ml#xpointer(string-range(//main,"衆参両院 ")[1])</simple\_ont1:context> <simple\_ont1:conventional\_form>参院 </simple\_ont1:conventional\_form>

</simple\_ont1:ORGANIZATION>

# 2.2. Semantic Nesting

Overlapping structures may occur within NE+s due to both syntactic and/or semantic reasons. In the case of local syntactic structures such as conjunctions we have provided guidelines to help disambiguate such structures and map them to conventional forms. We have not yet said anything about semantic overlapping regions which occur where NE+s. Explicit annotation of their relationships will sometimes be necessary where rich annotation of a text must take place.

For example, the term we see in (14) is annotated in the usual way in (15) showing that the term belongs to the class of DNA. There is also another term "human immunodeficiency virus" that belongs to the class VIRUS as shown in (16) which is completely contained inside the parent term. This hierarchical relation is made explicit by the use of the *constituents* property in (15) which points to the label of (16).

Ex. 14. human immunodeficiency virus long terminal repeat

Ex.15.<biol:DNA rdf:about="&bio1;00022" <bio1:context>http://www.biojournal.com/page.html# xpointer(string-range(//main,"human immunodeficiency virus long terminal repeat")[1])</bio1:context> <bio1:conventional\_form> human immunodeficiency virus long terminal repeat</bio1:conventional\_form> <bio1:constituents>&bio1:00023</bio1:constituents

>

</biol:DNA>

Ex.16.<bio1:VIRUS rdf:about="&bio1;00023" <bio1:context>http://www.biojournal.com/page.html# xpointer(string-range(//main,"human immunodeficiency virus")[1])</bio1:context> <bio1:conventional\_form> human immunodeficiency virus</bio1:conventional\_form>

</bio1:VIRUS>

The PIA guidelines themselves make no stipulation at this time about whether nesting of semantic annotations should be implemented or not, they simply offer the option to do so. If used, nesting will need to be governed by the domain ontology and this will require careful thought so that the knowledge can be used in a meaningful way. The danger of course is that without planning it will be difficult for annotators to maintain consistency with each other. In the extreme case the safest option is to forbid overlapping regions altogether and insist on a 'flat' style of annotation scheme.

# 3. Quality assurance

Our most basic aim is to produce annotated texts that are useful for machine learning so that tools can be trained to aid users in finding information quickly and reliably. Quality assurance is therefore a key aim of these guidelines and will only occur when there is agreement and consistency among the annotators in the domain. We consider that there are several co-related points to ensuring quality of annotation.

- 1 A tool to support annotation and ontology creation.
- 2 The use of the *source*, *created* and *modified* attributes to maintain versions of annotations.
- 3 Post-verification of annotations through the use of *sure* and *comment* properties.
- 4 Support for the correction of broken links through the use of the *orphan* property.
- 5 A measure of annotation consistency.
- 6 Annotator training and written guidelines.

In this paper we have concentrated on the presentation of the underlying theory and practice of an annotation scheme that links instances to concepts to an ontology for use on the Semantic Web. We expect that the first stage of the annotation task will be for the domain community to create an ontology that reflects the concepts used in the texts they want to describe. Clearly what is also needed is tool support to enable the creation of ontologies and instance capturing that is consistent with these guidelines. This is covered in Section 4.

The use of *source* attributes enables different versions of a document's annotation to co-exist. These versions could be produced by different sources one of which is considered to be the most trusted: the 'Gold Standard'. In PIA we consider that because of the cost only a few texts will be annotated to the Gold Standard by a domain expert and that these will be used to train a computer program to annotate other texts to a 'near-to Gold Standard'.

There needs to be some way to compare annotations provided by less trusted sources (such as a computer program) to the 'gold standard' benchmark in a quantitative way. Such a comparison is very useful to find points of confusion or disagreement between annotators and to make revisions to annotation policy guidelines or computer programs that perform the annotation<sup>4</sup>. At least two methods are possible: The first is the MUC-style scoring mechanism (Chinchor & Dungca, 1995) which is phrase-based and in which separate points are awarded for successfully finding the NE boundary and the NE class. The second is word-based scoring in a similar manner to POS evaluation in which the word class from the less trusted source is compared to the class from the trusted source and points are calculated for recall and precision for each class, see for example (van Halteren et al., 2001).

Finally and perhaps most important is annotator training. Before embarking on the creation of a large corpus it is necessary to consider training annotators so that they have the same view of the domain. This involves annotating the same texts, comparing and discussing results and agreeing a common policy for difficult cases which should be written down as a supplement to these guidelines to help ensure consistency. Although it is likely that annotators will share the same intuition about the domain knowledge structure, it is unlikely that they will agree at the beginning on how to 'decode' surface text into this knowledge structure. It is the task of these guidelines to aid this process but this will need to be supplemented by case studies for each domain.

# 4. Tool Support

It is necessary for any knowledge annotation scheme to provide software support for annotators and PIA is no different in this respect. There are already several tools that support ontology creation and maintenance that can be used to provide part of the knowledge creation support that we need. Of these only a few provide support for the Semantic Web model such as Ontomat (Handschuh et al., 2001) (University of Karlsruhe) produced as part of the OntoAgent project (Staab et al., 2000) and Protégé-2000 (Fridman et al., 2000) (Musen et al., 2000) (Stanford University). Ontology creation is only part of our requirement and both these tools provide additional features that are useful for our purpose such as instance capturing and saving of data in Semantic Web languages. Currently only Ontomat provides support for capturing instances directly from texts using a convenient highlight and drag-and-drop graphical user interface (GUI) although it does not currently support linkage back from the instance to its occurrence in the text. Instances in Protégé-2000 must be entered by hand and there is currently no support for semi-automatic instance capturing.

The first version of the PIA Annotation Tool (PAT v1.0) has been built in Java as a plug-in to Protégé-2000. This allows users to annotate documents in a simple dragand-drop manner according the ontology including the specification of coreference links. The linkage system used in this first version is a simple byte pointer mechanism, whereas in future versions we will be upgrading this to the XPointer mechanism we described here. PAT v1.0 is available from the PIA Web site at http://research.nii.ac.jp/PIA/.

# 5. Conclusion

In this paper we have given a concise description of guidelines which are suitable for annotating a wide range of terminological and name expressions in texts according to a given ontology that are suitable for communicating knowledge on the Semantic Web.

Although the notation is not ideal we feel that it provides reasonable power of expression and good consistency. Compared to the MUC-7 guidelines we can now support discontinuous NEs and at the same time explicitly record the relationships between NE classes through the ontology. The additional cost is in the formulation of models that must consider how to recover the conventional forms of NEs. We feel though that the result should be to make explicit all of the objects that are being referred to by the author which is what we need for higher levels of information extraction.

It is also worth noting that the use of an ontology provides several advantages, not only in making relations between NE+s explicit, but in the support for logical and reasoning, and also the possibility of using the ontology as a 'pivot' between languages since it is perfectly possible and natural to use the same ontology as the basis for annotation in a multilingual document collection.

Finally we must emphasize that the guidelines in themselves only offer half a solution to annotation. It is

<sup>&</sup>lt;sup>4</sup> See (Kehler et al., 2001) for a discussion of these issues.

expected that in each domain, the guidelines will be supplemented by case studies and protocols such as whether semantic nested structures are allowed or not.

### 6. Acknowledgements

While any mistakes or errors are entirely our own, we would like to express our thanks to a number of our colleagues for useful discussions that arose in the course of drafting the first version of the guidelines and who offered comments and further references: these include

# 7. References

- T. Berners-Lee, M. Fischetti, and M. Dertouzos. (1999). Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web. Harper, San Francisco, September. ISBN: 0062515861.
- B. de Besse, B. Nkwenti-Azeh, and J. C. Sager. (1997). Glossary of terms used in terminology. Terminology 4(1), pages 117-156. John Benjamins, Amsterdam.
- D. Brickley and R.V. Guha (eds.), (2000). Resource Description Framework (RDF) Schema Specification 1.0. CR, W3C, Mar. 2000. <u>http://www.w3.org/TR/2000/CR-rdf-schema-20000327</u>.
- N. Chinchor and G. Dungca. (1995). Four scorers and seven years ago: The scoring methods for MUC-6. In Proceedings of the Sixth Message Understanding Conference (MUC-5), Baltimore, Maryland, USA., pages 33–38. Morgan Kaufmann, November.
- N. Chinchor, editor. (1997). MUC-7 Named Entity Task Definition, Version 3.5, This document should be available from Nancy Chinchor <u>chinchor@gso.saic.com</u>, September 17th. DARPA.
- N. Collier, K. Takeuchi, K. Tsuji, G. Jones, J. Fukumoto, N. Ogata, C. Nobata, and K. Ono. (2001). The PIA project: Learning to semantically annotate texts form an ontology and XML-instance data. In position paper proceedings of the International Semantic Web Working Symposium (SWWS), Stanford University, California, USA, pages 8–9, 30th July – 1st August.
- N. Collier and K. Takeuchi, (2002). PIA-Core: Semantic Annotation through Example-based Learning. In proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002), Las Palmas, Spain, 29-31 May.
- DeRose, S., Maler, E. and Daniel, R. (eds), 2002. XML Pointer Language (XPointer) Version 1.0, W3C Candidate Recommendation 11 September 2001. Available from http://www.w3.org/TR/xptr/
- Dublin Core, (1999). Dublin Core Metadata Element Set, Version 1.1: Reference Description. Technical report, Dublin Core Metadata Initiative, Jul. 1999. <u>http://purl.org/DC/documents/rec-dces-19990702.htm</u>.
- N. Fridman Noy, R. W. Fergerson, and M. A. Musen. (2000). The knowledge model of Protégé-2000: combining interoperability and flexibility. In Proceedings of the 2nd International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000), Juan-les-Pins, France, pages 1–20, 2000.
- R. Grishman (1998), TIPSTER Text Architecture Design Version 3.1, NIST Technical Report of the TIPSTER Project, 7<sup>th</sup> October 1998.

Beatrice Daille (IRIN, University of Nantes, France), Satoshi Sekine (University of New York, USA) and Keita Tsuji (NII, Japan).

- S. Handschuh, S. Staab, and A. Maedche. (2001). CREAM –creating relational metadata with a component-based, ontology-driven annotation framework. In First International Conference on Knowledge Capture (K-CAP 2001), Victoria, B.C., Canada, 21 – 23 October.
- N. Ide and G. Priest-Dorman, editors. (2000). Corpus Encoding Standard. Number CES 1, Version 1.5. Expert Advisory Group on Language Engineering Standards (EAGLES), http://www.cs.vassar.edu/CES/, March.
- Kahan, J., Koivunen, M., Prud'Hommeaux, E. and Swick, R., 2001. Annotea: An Open RDF Infrastructure for Shared Web Annotations, in proceedings of the 10th International World Wide Web Conference, Hong Kong, May 2001.

http://www10.org/cdrom/papers/488/index.html.

- A. Kehler, J. Bear, and D. Appelt. (2001). The need for accurate alignment in natural language system evaluation. Computational Linguistics, 27(2):231–248, June.
- O. Lassila and R. Swick (eds.), (1999). Resource Description Framework (RDF) Model and Syntax Specification. Recommendation, W3C, Feb. 1999. <u>http://www.w3.org/TR/1999/REC-rdf-syntax</u>.
- M. A. Musen, R. W. Fergerson, W. E. Grosso, N. F. Noy, M. Crube'zy, and J. H. Gennari. (2000). Componentbased support for building knowledge-acquisition systems. In Conference on Intelligent Information Processing (IIP'2000) of the International Federation for Processing World Computer Congress (WCC'2000), Beijing.
- S. Staab, M. Erdmann, A. Maedche, and S. Decker. (2000). An extensible approach for modeling ontologies in RDF(S). In Proceedings of Workshop on the Semantic Web, held at the Fourth European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2000), Lisbon, Portugal.
- H. van Halteren, J. Zavrel, and W. Daelemans. (2000). Improving accuracy in word class tagging through the combination of machine learning systems. Computational Linguistics, 27(2):199–229, June 2001.