Stretching the TEI: Converting the Genia Corpus

Tomaž Erjavec
Department of Intelligent Systems
Jožef Stefan Institute, Ljubljana

Jin-Dong Kim
CREST
Japan Science and
Technology Corporation

Tomoko Ohta
Department of Information Science
University of Tokyo

Yuka Tateisi
CREST JSTC

Jun-ichi Tsujii
CREST JSTC

Abstract

The paper discusses the application of the Text Encoding Initiative Guidelines to a linguistically annotated corpus. The recently released GENIA corpus Version 3.0 contains 2,000 abstracts taken from the MEDLINE database, and has almost 100,000 hand-annotated terms, which are marked for semantic class from the accompanying ontology. The paper introduces and shows how to apply the TEI, which has become a defacto standard in corpus encoding, to this corpus. It overviews the history of TEI, including recent and expected developments, and then turns to implementing a TEI parametrisation and conversion for the GENIA corpus. Discussed are some problems and choices that arise in this process.

1 Introduction

Text mining from biological literature is emerging as one of the main issues in bioinformatics research, a huge and thriving field. Natural language processing methods could significantly raise the potential of utilising this literature, with applications ranging from intelligent searches to automatic discovery of scientific theories. Yet, while NLP techniques are relatively domain-portable, reference materials, e.g., corpora, are not. The lack of a large annotated corpus of biological texts can thus be seen as a major bottleneck for applying NLP techniques to bioinformatics. This was the reason behind the compilation of the GENIA corpus (Ohta et al., 2002).

In this paper we show how to develop a standardised encoding for such a resource, and for others of its kind. For this we use the Text Encoding Initiative Guidelines P4 (Sperberg-McQueen and Burnard, 2002), and specify a constructive mapping, i.e., an XSLT stylesheet, to the developed encoding.

The motivation for this re-encoding is that TEI is well-designed and widely accepted architecture, which has been often used for annotating language corpora, and by porting to it, GENIA can gain new insights into possible encoding practices and maybe make the corpus better suited for interchange. As the transformation to TEI is fully automatic, there is also no need to abandon the original markup format of GENIA, which, as it has been crafted specially for the corpus, provides a tighter encoding than can be possible with the more general TEI.

The paper thus proposes the creation of a practical annotation scheme for linguistically annotated (biomedical) corpora, the conversion to which is automatic and supports consistency checking and validation. The paper also serves as a guide to parametrising TEI and overviews its modules that might be useful for encoding linguistically annotated corpora; here we also discuss the shortcomings and expected developments of these modules.

The paper is structured as follows: Section 2 introduces the GENIA corpus; Section 3 turns to the

TEI and gives its history, pros and cons of using it, and the method of parametrising TEI for particular projects; Section 4 discusses such a parametrisation for GENIA; Section 5 reviews the structure of the corpus, giving in parallel the original and the TEI encodings and explains the conversion of the corpus to TEI; finally, Section 6 offers some conclusions and directions for further work.

2 The GENIA Corpus

The GENIA corpus (Ohta et al., 2002) is being developed in the scope of the GENIA project, which seeks to develop information extraction techniques for scientific texts using NLP technology. The corpus consists of semantically annotated published abstracts from the biomedical domain. The corpus is a collection of articles extracted from the on-line MEDLINE abstracts (U.S. National Center for Biotechnology Information, *http://www.ncbi.nlm.nih.gov/*, PubMed database). Since the focus of the corpus is on biological reactions concerning transcription factors in human blood cells, articles were selected that contain the MeSH terms *human*, *blood cell* and *transcription factor*.

For those not familiar with the field we should note that the articles are composed largely of structurally very complex technical terms, and are almost incomprehensible to a layperson. A typical heading e.g., reads *IL-2 gene expression and NF-kappa B activation through CD28 requires reactive oxygen production by 5-lipoxygenase.*

The main value of the GENIA corpus comes from its annotation: all the abstracts and their titles have been marked-up by two domain experts for biologically meaningful terms, and these terms have been semantically annotated with descriptors from the GENIA ontology.

The GENIA ontology is a taxonomy of, currently, 47 biologically relevant nominal categories, such as *body part*, *virus*, or *RNA domain or region*; the taxonomy has 35 terminal categories.

The terms of the corpus are semantically defined as those sentence constituents that can be categorised using the terminal categories from the ontology. Syntactically such constituents are quite varied: they include qualifiers and can be recursive.

The GENIA corpus is encoded in the Genia Project Markup Language. The GPML is an XML DTD (Kim et al., 2001) where each article contains its MEDLINE ID, title and abstract. The texts of the abstracts are segmented into sentences, and these contain the constituents with their semantic classification. Examples will be given in Section 5.

The GENIA ontology is provided together with the GENIA corpus and is encoded in DAML+OIL (*http://www.daml.org/*), the standard XML-based ontology description language.

A suite of supporting tools has been developed or tuned for the GENIA corpus and GPML: the term annotation is performed with the XMLMind editor; an XPath-based concordancer has been developed for searching the corpus; and CSS stylesheets are available for browsing it.

The GENIA corpus V2.1 has been released in August 2002 and is the prototype version: it contains 670 abstracts (cca 160,000 words) annotated for terms, and also tokenised and marked for PoS. PoS tagging has been performed automatically and later — to an extent — hand validated. In this version, the GPML DTD and resource organisation was also more complex than presented above: each article could contain local resources that included the article-specific ontology as well as a lexicon, which mediated between the text and the ontology.

In December 2002, Version 3.0 has been released. It consists of 2,000 abstracts with over 400,000 words and more than 90,000 marked-up terms. The structure has also been simplified, without the local resources. This version has not yet been marked-up with tokens or PoS information.

The GENIA corpus is available free of charge from the GENIA project homepage, at *http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/*.

3 The Text Encoding Initiative

The Text Encoding Initiative was established in 1987 as a systematised attempt to develop a fully general text encoding model and set of encoding conventions based upon it, suitable for processing and analysis of any type of text, in any language, and intended to serve the increasing range

of existing (and potential) applications and uses. The TEI Guidelines for Electronic Text Encoding and Interchange, were first published in April 1994 in two substantial green volumes, known as TEI P3. In May 1999, a revised edition of TEI P3 was produced, correcting several typographic and other errors. In December 2000 the TEI Consortium (*http://www.tei-c.org/*) was set up to maintain and develop the TEI standard; the Consortium is managed by a Board of Directors, and its technical work is overseen by an elected Council.

In 2002, the TEI Consortium announced availability of a major revision of TEI P3, the TEI P4 (Sperberg-McQueen and Burnard, 2002) the object of which is to provide equal support for XML and SGML applications using the TEI scheme. The revisions needed to make TEI P4 have been deliberately restricted to error correction only, with a view to ensuring that documents conforming to TEI P3 will not become illegal when processed with TEI P4. For GENIA, we are using the XML-compatible version of TEI P4.

In producing P4, many possibilities for other, more fundamental changes have been identified. With the establishment of the TEI Council, it became possible to agree on a programme of work to enhance and modify the Guidelines more fundamentally over the coming years. TEI P5 will be the next full revision of the Guidelines. The work on P5 has now started, and the date of its appearance will likely be in 2004. There are currently several TEI Working Groups addressing various parts of the Guidelines that need attention; some of these will be mentioned in the following sections.

More than 80 projects spanning over 30 languages have so far made use of the TEI guidelines, producing such diverse resources as computer corpora, medieval literature, text-critical editions of classical works, dictionaries, and library catalogues. TEI has also been influential in corpus encoding, where the best known example is probably the British National Corpus. However, while the TEI has been extensively used for annotating PoS tagged corpora, it been less popular for encoding texts used by the the Information Retrieval/Extraction community; there, a number of non-TEI initiatives have taken the lead in encoding, say, ontologies or inter-document linking. As will be seen, the content and original encoding of GENIA pose a number of challenges in the conversion to TEI.

3.1 Pros and cons of using TEI

Why, if a corpus is already encoded in XML using a home-grown DTD, to re-encoded it in TEI at all? One reasons is certainly the validation aspect of the exercise: re-coding a corpus, or any other resource, reveals hidden (and in practice incorrect) assumptions about its structure. Re-coding to a standard recommendation also forces the corpus designers to face issues which might have been overlooked in the original design.

There are also other advantages of using TEI as the interchange format: (1) it is a wide-coverage, well-designed (modular and extensible), widely accepted and well-maintained architecture; (2) it provides extensive documentation, which comprises not only the Guidelines but also papers and documentation (best practices) of various projects; (3) it offers community support via the tei-l public discussion list; (4) various TEI-dedicated software already exists, and more is likely to become available; and (5) using it contributes to the adoption of open standards and recommendations.

However, using a very general recommendation which tries to cater for any possible situation brings with it also several disadvantages:

Tag abuse TEI might not have elements / attributes with the exact meaning we require. This results in a tendency to misuse tags for purposes they were not meant for; however, it is a case of individual judgement to decide whether to (slightly) abuse a tag, or to implement a local extension to add the attribute or element required.

Tag bloat Being a general purpose recommendation, TEI can — almost by definition — never be optimal for a specific application. Thus a custom developed DTD will be leaner, have less (redundant) tags and simpler content models.

TEI for humanities While the Guidelines cover a vast range of text types and annotations, they are maybe the least developed for "high

level" NLP applications or have failed to keep abreast of "cutting-edge" initiatives. As will be seen, critical areas are the encoding of ontologies, of lexical databases and of feature structures.

3.2 Building a TEI DTD

The TEI Guidelines (Sperberg-McQueen and Burnard, 2002) consist of the formal part, which is a set of SGML/XML DTD fragments, and the documentation, which explains the rationale behind the elements available in these fragments, as well as giving overall information about the structure of the TEI.

The formal SGML/XML part of TEI comes as a set of DTD fragments or tagsets. A TEI DTD for a particular application is then constructed by selecting an appropriate combination of such tagsets. TEI distinguishes the following types of tagsets: (1) the **core tagset**, which is always included without any special action by the encoder; (2) the **base tagsets**, which are the basic building blocks for specific text types; exactly one base must be selected by the encoder, unless one of the combined bases is used; (3) the **additional tagsets**, which define extra tags useful for particular purposes; all additional tagsets are compatible with all bases and with each other; (4) **user defined tagsets**, which give the possibility of extending and overriding the definitions provided in the TEI tagset.

While a project-particular XML DTD can be constructed by including and ignoring the TEI DTD fragments directly, it is also possible for easier processing to build a one-file DTD with the help of the on-line TEI Pizza Chef service, available from the TEI web site.

4 Parametrising TEI for GENIA

This section gives our proposal on how to encode GENIA, and in general, ontology-annotated corpora, in the TEI. A number of tagsets could prove useful in the long term, and we have chosen a parametrisation of TEI that collects not only those that we consider necessary for the current version of GENIA, but also some that might prove of service in the future. Furthermore, we support the encoding of both version 2.1 and 3.0 of the corpus. The resulting DTD is thus very generous in what kinds of data it caters for. We give in Figure 1 the XML prolog of the TEI encoded corpus that defines our parametrisation of TEI.

```
<!DOCTYPE teiCorpus.2 SYSTEM
 "http://www.tei-c.org/P4X/DTD/tei2.dtd"
 [<!ENTITY % TEI.prose        "INCLUDE">
  <!ENTITY % TEI.general      "INCLUDE">
  <!ENTITY % TEI.dictionaries "INCLUDE">
  <!ENTITY % TEI.terminology  "INCLUDE">
  <!ENTITY % TEI.linking      "INCLUDE">
  <!ENTITY % TEI.analysis     "INCLUDE">
  <!ENTITY % TEI.fs           "INCLUDE">
  <!ENTITY % TEI.corpus       "INCLUDE">
  <!ENTITY % TEI.XML          "INCLUDE">
  <!ENTITY % TEI.extensions.ent SYSTEM
            'geniaex.ent'>
  <!ENTITY % TEI.extensions.dtd SYSTEM
            'geniaex.dtd'>
 ]>
```

Figure 1: The XML TEI prolog for GENIA

4.1 TEI.prose

The base tagset does not declare many elements but rather inherits all of the TEI core, which includes the TEI header, and text elements. A TEI document will typically have as its root element ⟨*TEI.2*⟩ which is composed of the ⟨*teiHeader*⟩, followed by the ⟨*text*⟩.

The TEI header describes an encoded work so that the text (corpus) itself, its source, its encoding, and its revisions are all thoroughly documented. TEI.prose also contains elements and attributes for describing text structure, e.g. ⟨*div*⟩ for text division, ⟨*p*⟩ for paragraph, ⟨*head*⟩ for text header, etc. The tagset is therefore useful for encoding the gross structure of the corpus texts.

4.2 TEI.general

This combined base allows the combination of base tagsets, with the proviso, that each appear within its own division, ⟨*div*⟩. We use it to circumvent the requirement that a TEI DTD should contain only one base tagset.

This option was necessary for the V2.1 version of the GENIA corpus, where each article could contain not only the text of the abstract but also a local lexicon and ontology, as each of these is modelled using a different base tagset, as is explained next.

4.3 TEI.dictionaries

This base tagset is oriented toward printed dictionaries. While the GENIA lexicon for Version 2.1 (Version 3.0 does not include a lexicon) is significantly different from a printed dictionary, this tagset does offer, at the current depth of encoding, elements which are suitable for expressing the desired lexical markup. In particular, each lexical entry is encoded in the ⟨*entry*⟩ and the form of the entry in the ⟨*form*⟩ elements.

However, this is only a stop-gap measure. As TEI does not have any working group devoted to lexica, it might be better to look for an lexicon interchange encoding further afield, say to the Open Lexicon Interchange Format, *http://www.olif.net/*. Another option, which directly builds on the TEI, is the CONCEDE Lexica Database Model (Erjavec et al., 2000).

Version 3.0 of the corpus does not use lexica, so this issue has been, for the time being, put aside.

4.4 TEI.terminology

This base tagset is used for encoding terminological databases, which we, for V2.1, used to encode local ontologies; this is the closest TEI comes to offering a base useful for encoding general ontologies.

As is noted in the P4 Guidelines themselves, the TEI chapter on encoding terminology has been rendered obsolete in several respects, chiefly as a result of the publication of MARTIF, the ISO 12200:1999 standard "Machine-readable terminology interchange format".

Version 3.0 of the corpus does, in any case, not use local ontologies. Furthermore, the structure of the ontology has been in V3.0 explicated, which is why we now use a simpler encoding that stores it the corpus TEI header, as will be discussed in Section 5.4.

4.5 TEI.corpus

This additional tagset introduces a new root element, ⟨*teiCorpus.2*⟩, which comprises a (corpus) header and a series of ⟨*TEI.2*⟩ elements. The TEI.corpus tagset also extends the certain header elements to provide more detailed descriptions of the corpus material.

4.6 TEI.linking

This additional tagset provides mechanisms for linking, segmentation, and alignment. The elements provided here enable links to be made e.g., between the articles and their source URLs, or between concepts and their hypernyms.

It should be noted that while the TEI treatment of external pointers had been very influential, it was overtaken and made obsolete by newer recommendations. However, the TEI does have a Working Group on Stand-Off Markup, XLink and XPointer, which should produce new TEI encoding recommendations for this area in 2003.

4.7 TEI.analysis

This additional tagset is used for associating simple linguistic analyses and interpretations with text elements. It can be used to annotate words, ⟨*w*⟩, clauses, ⟨*cl*⟩, and sentences, ⟨*s*⟩ with dedicated tags, as well as arbitrary and possibly nested segments with the ⟨*seg*⟩. Such elements can be, via attributes, associated with their analyses. This tagset has proved very popular for PoS-annotated corpora.

4.8 TEI.fs

This additional tagset is used to mark-up the text with feature structures. In addition, the TEI Feature Structure Declaration is provided, which is an auxiliary DTD, for defining feature values and names, their descriptions and constraints on valid feature structures.

While the current versions of the GENIA corpus do not use elements from this tagset, we included it for future reference. Namely, the corpus should eventually have markup for deep syntactic analyses, possibly in the HPSG framework. For this, a feature-structure encoding is necessary, and the TEI tagset offers a venue for experimentation.

The current TEI proposal has some disadvantages; foremost, there are no known application of this tagset we could find. The proposal is also tailored toward GPSG rather than HPSG and has no support for a type hierarchy or co-indexing. Finally, there are no mechanisms (apart from the DTD) for checking validity of specified feature structures.

But there also exist reasons for pursuing the possibility of using TEI for encoding feature-structures. First, there are, to our knowledge, no other (much less better) standardised efforts to encode them. Also, there exists now a number of unification-based parsers with large grammars and lexica, and the beginnings of feature-structure annotated corpora (e.g., BulTreeBank), so the field might be ready to start exchanging the resources. Finally, the TEI.fs is being taken forward under the auspices of the ISO Technical Committee 37 (Terminology and Other Language Resources). Working Group 4 of TC37 (Language Resource Management) is in the process of forming a group to oversee the development of TEI.fs into an international standard.

4.9 TEI.XML

TEI P4 allows both standard SGML and XML encodings. Including the TEI.XML option indicates that the target DTD is to be expressed in XML.

4.10 TEI.extensions.ent

The file gives, for each element sanctioned by the chosen modules, whether we include or ignore it in our parametrisation. While this is not strictly necessary (without any such specification, all the elements would be included) we thought it wise to constrain the content models somewhat, to reduce the bewildering variety of choices that the TEI otherwise offers. Also, such an entity extension file gives the complete list of all the TEI elements that are allowed (and disallowed) in GENIA, which might prove useful for documentation purposes.

4.11 TEI.extensions.dtd

This file specifies the changes we have made to TEI elements. We have e.g., added the *url* attribute to ⟨*xptr*⟩ and ⟨*xref*⟩ and tagging attributes to word and punctuation elements.

5 GENIA in TEI

With the TEI DTD in place, it is possible to specify the mapping between the original GPML encoding and the TEI one. Formally, the mapping is an XSLT stylesheet, as further discussed below. Here we present this translation by giving examples of the input and output encodings and comparing the two.

5.1 Corpus structure

As shown in Figure 2, the most noticeable difference between GPML and TEI is, apart from the renaming of elements, the addition of headers to the corpus and texts. In the GENIA ⟨*teiHeader*⟩ we give e.g., the name, address, availability, sampling description, and, for each abstract's ⟨*sourceDesc*⟩, two ⟨*xptr*⟩s: the first gives the URL of the HTML article in the MEDLINE database, while the second is the URL of the article in the original XML. It should be noted that we use a locally defined *url* attribute for specifying the value of the pointer.

In V2.1 the local resources of the article, namely the ontology and the lexicon are in the TEI encoding contained in two separate ⟨*div*⟩ elements; this is not used in V3.0.

5.2 Term annotation

As shown in Figure 3, the text of the abstract is first analysed into sentences, and these into constituents (terms), which use an attribute to point to the appropriate GENIA ontology class.

The main difference in the two encodings, apart from renaming, are the uses of the attributes *ana* and *function* on clauses. Whereas in GPML, the *sem* attribute can hold either the pointer to the semantic class, or an expression, it is in TEI the *ana* that holds the `#IDREF`, while *function* contains the complex expressions.

5.3 Token annotation

For V1.1 we have also annotated the GPML version of the corpus with LTG tools (Grover et al., 2002). In short, the corpus is tokenised, and then part-of-speech tagged with two taggers, each one using a different tagset, and the nouns and verbs lemmatised. Additionally, the deverbal nominalisations are assigned their verbal stems.

The conversion to TEI is also able to handle this additional markup, by using the TEI.analysis module. The word and punctuation tokens are encoded as ⟨*w*⟩ and ⟨*c*⟩ elements respectively, which are further marked with *type* and *lemma* and the locally defined *c1*, *c2* and *vstem*.

```
<!DOCTYPE set SYSTEM "gpml.dtd">
<set>
 <article>
  <articleinfo><bibliomisc>
     *MEDLINE_ID*
   </bibliomisc></articleinfo>
  <title>
   *Title_of_article*
  </title>
  <abstract>
   *Abstract_of_article*
  </abstract>
  <localresource> <!--V2.1-->
   <imports *Ontology REF*"/>
   *Local_ontology*
   *Local_lexicon*
  </localresource>
 </article>
 *More_articles*
</set>
```

```
<!DOCTYPE teiCorpus.2 SYSTEM "genia-tei.dtd">
 <TEIcorpus.2>
  <teiHeader type="corpus">
   *Corpus_header*</teiHeader>
   <TEI.2 id="*MEDLINE_ID*">
     <teiHeader type="text">
      *Article_header*</teiHeader>
    <text><body>
     <div type="abstract">
      <head>*Title_of_article*</head>
      <p>*Abstract_of_article*</p>
     </div>
     <div type="ontology"> <!--V2.1-->
      *Local_ontology/TEI.terminology*
     </div>
     <div type="lexicon">  <!--V2.1-->
      *Local_lexicon/TEI.dictionaries*
     </div>
    </body></text></TEI.2>
   *More_articles*</TEIcorpus.2>
```

Figure 2: The GPML and TEI structure of the corpus

```
<cons sem="(AND G#other_name G#other_name)">
 <cons>Cellular</cons> and
 <cons>molecular</cons>
 <cons>mechanisms</cons>
</cons> of
<cons sem="G#other_name">IFN-gamma
production</cons> induced by ...
```

```
<cl function="(AND G.other_name
  G.other_name)" ana="G.other_name">
  <cl>Cellular</cl> and <cl>molecular</cl>
  <cl>mechanisms</cl>
</cl> of
<cl ana="G.other_name">IFN-gamma
production</cl> induced by...
```

Figure 3: The GPML and TEI encoding of terms

5.4 The ontology

One of the more interesting questions in recoding GENIA in TEI was how to encode the ontology; in V2.1 it could be included in the local resources and this was modelled in TEI.terminology. However, this choice has proved too complex and out of touch with current practices. As shown in the left side of Figure 4 the ontology is in V3.0 encoded in a separate document, conforming to the OIL+DAML specification. This, inter alia, means that that XML file heavily relies on XML Namespaces and the RDF recommendation.

As currently the GENIA ontology can be modelled by a taxonomy, we have now translated it to the TEI ⟨*taxonomy*⟩ element, which is contained in the ⟨*classDecl*⟩ of the header ⟨*encodingDesc*⟩. The TEI defines this element as "[the classification declaration] contains one or more taxonomies defining any classificatory codes used elsewhere in the text", i.e., exactly suited for our purposes.

There are quite substantial differences between the two encodings: the DAML+OIL models class inclusion with links, while the TEI does it as XML element inclusion. This is certainly the simpler and more robust solution, but requires that the ontology is a taxonomy, i.e., tree structured. The second difference is in the status of the identifiers: in DAML+OIL they are general `#CDATA` links, which need a separate (XLink/XPointer) mechanisms for their resolution. In TEI they are XML `ID` attributes, and can rely on the XML parser to resolve them. While this is a simpler solution, it does support document-internal reference only.

5.5 Conversion of GPML to TEI

Because the source format of GENIA will remain the simpler GPML, it is imperative to have an automatic procedure for converting to the TEI inter-

```
<daml:Class rdf:ID="source"></daml:Class>
<daml:Class rdf:ID="natural">
 <rdfs:subClassOf rdf:resource="#source"/>
</daml:Class>
<daml:Class rdf:ID="organism">
 <rdfs:subClassOf rdf:resource="#natural"/>
</daml:Class>
<daml:Class rdf:ID="multi_cell">
 <rdfs:subClassOf rdf:resource="#organism"/>
</daml:Class>
...
```

```
<taxonomy id="G.taxonomy">
 <category id="G.source">
  <catDesc>biological source</catDesc>
  <category id="G.natural">
   <catDesc>natural</catDesc>
   <category id="G.organism">
    <catDesc>organism</catDesc>
    <category id="G.multi_cell">
     <catDesc>multi-cellular</catDesc>
    </category>
...
```

Figure 4: The GENIA DAML+OIL and TEI ontology

change format. The translation process takes advantage of the fact that both the input and output are encoded in XML, which makes it possible to use the XSL Transformation Language, XSLT that defines a standard declarative specification of transformations between XML documents. There also exist a number of free XSLT processors; we used Daniel Veillard's `xsltproc`.

The transformation is written as a XSLT stylesheet, which makes reference to two documents: the GENIA ontology in TEI and the template for the corpus header. The stylesheet then resolves the GPML encoded corpus into TEI. The translation of the corpus is thus fully automatic, except for the taxonomy, which was translated by hand.

6 Conclusions

The paper proposed a TEI encoding for GENIA and specified a mapping from the Genia markup language to this encoding. The conversion has been implemented in XSLT, and both the PoS marked up version with local resources, as well as the larger but structurally simpler version 3.0 have been translated to our XML paramterisation of TEI P4. The parametrisation (DTD) and the XSLT stylesheets are, together with a report document them, available at *http://nl.ijs.si/et/genia/*.

We have attempted to survey the TEI modules that can be useful for encoding a wide variety of linguistically annotated corpora and to comment on the areas of the TEI where the Guidelines need attention. The paper, it is hoped, can thus serve as a blueprint for parametrising TEI for diverse corpus resources.

As for GENIA, the corpus should in the future gather more complex annotations, say for chunking and parsing. Interesting is also the inclusion of other knowledge sources into the corpus, say of Medical Subject Headings (MeSH), Unified Medical Language System (UMLS), International Classification of Disease (ICD), etc. The place of these annotations in the corpus will have to be considered, and their linking to the existing information determined.

References

Tomaž Erjavec, Roger Evans, Nancy Ide, and Adam Kilgarriff. 2000. The Concede Model for Lexical Databases. In *Second International Conference on Language Resources and Evaluation, LREC'00*, pages 355–362, Paris. ELRA.

Claire Grover, Ewan Klein, Alex Lascarides, and Maria Lapata. 2002. XML-based NLP Tools for Analysing and Annotating Medical Language. In *2nd Workshop on NLP and XML (CoLing Workshop NLPXML-2002). http://www.ltg.ed.ac.uk/software/ttt/*.

Jin-Dong Kim, Tomoko Ohta, and Jun-ichi Tsujii. 2001. XML-based linguistic annotation of corpus. In *Proceedings of the first NLP and XML Workshop*, pages 44–53.

Tomoko Ohta, Yuka Tateisi, and Jin-Dong Kim. 2002. The GENIA Corpus: an Annotated Research Abstract Corpus in Molecular Biology Domain. In *Proceedings of the Human Language Technology Conference*, to appear.

C. M. Sperberg-McQueen and Lou Burnard, editors. 2002. *Guidelines for Electronic Text Encoding and Interchange, The XML Version of the TEI Guidelines*. The TEI Consortium. *http://www.tei-c.org/*.