

BOOTStrep Annotation Scheme: Encoding Information for Text Mining

Scott Piao,¹ Ekaterina Buyko,² Yoshimasa Tsuruoka,¹
Katrin Tomanek,² Jin-Dong Kim,³ John McNaught,¹
Udo Hahn,² Jian Su⁴ and Sophia Ananiadou¹

Abstract

Annotation of information in corpora is an important aspect of text mining. It bridges between the information hidden in natural language texts and the semantic search queries for the information desired by users. Due to the complex nature of the information needed for text mining, it is essential to design comprehensive annotation schemes to encode and organise the unstructured information contained in text into certain structured forms that can be processed by computers. In the EU BOOTStrep Project⁵, which aims to pull together existing biological databases and various terminological repositories and implement a text analysis system to populate a Bio-Lexicon and a Bio-Ontology to support text mining, an annotation scheme has been under construction for encoding multilayer information pertaining to text mining applications. In particular, it is aimed at supporting the interoperability among a set of text annotator tools. In practice, this annotation scheme is built into a type system of a software architecture, named the Unstructured Information Management System (UIMA), which provides a software architecture for the integration and interoperation of NLP annotation tools. Although this scheme is targeted at the biological domain and hence contains some biology-specific semantic elements, the major part of the annotation scheme covers generic language annotation, such as part-of-speech, syntactic parse, co-reference resolution, etc. Therefore, our scheme can easily be ported to other domains or generic language analysis tasks. Coupled with UIMA, this scheme provides a practical means of capturing information from texts for various purposes, such as text mining, NLP, corpus linguistics, etc.

1. Introduction

Information annotation in corpora is an important aspect of text mining. It bridges between the information hidden in natural language text and the semantic search queries for the information desired by users. Due to the complex nature of the information needed for text mining, comprehensive annotation schemes are essential

¹ School of Computer Science and National Centre of Text Mining, University of Manchester, UK
e-mail: {scott.piao;yoshimasa.tsuruoka;john.mcnaught;sophia.ananiadou}@manchester.ac.uk

² Language & Information Engineering Lab, Jena University, Germany
e-mail: {buyko;tomanek;hahn}@coling-uni-jena.de

³ Department of Computer Science, University of Tokyo, Japan
e-mail: jdkim@is.s.u-tokyo.ac.jp

⁴ Institute for Infocomm Research, Singapore
e-mail: sujian@i2r.a-star.edu.sg

⁵ The BOOTStrep Project (ref. 028099) is funded by the EC 6th Framework Programme. For further details, see the project website: <http://www.BOOTStrep.org>.

for extracting and organising the unstructured information conveyed by natural language into structured forms that can be processed automatically by computers.

In the EC BOOTStrep Project (refer to footnote 1), which aims to pull together existing biological databases and various terminological repositories and implement a text analysis system to populate a Bio-Lexicon and a Bio-Ontology to support bio-text mining (Ananiadou and McNaught, 2006), an annotation scheme has been designed to support multilayer information annotation pertaining to the text mining tasks. While there has been much previous work regarding annotation schemes, our work is different in that we aim to provide a comprehensive annotation scheme which supports a unified software architecture for the integration and interoperation of the annotation tools (in this paper, we constrain our discussion to the text annotation tools) via the Unstructured Information Management Architecture (UIMA) (Ferrucci and Lally, 2004).

The current scheme covers five layers of annotation: document meta information, document structure, morpho-syntactic categories, syntactic structure and semantic information. These categories correspond to the various types of annotation tools such as document metadata extractors, document structure analysis tools, part of speech (POS) taggers, syntactic parsers, named entity recognisers, etc. The annotation scheme provides a “common language” between the annotation tools for sharing the text and metadata. Although this scheme aims to cater for the needs of text mining in the biological domain, the major parts of the scheme, except the document metadata and semantic layer, stem from generic text annotations. It should be noted, however, that the implementation of the scheme in a practical software system for a specific domain involves adjusting and training the tools on domain-specific data. In the following sections, we will describe our scheme and discuss related issues.

2. UIMA annotation scheme and interoperability of annotation tools

The construction of the annotation scheme is motivated by the need of achieving interoperability for annotation tools under the UIMA architecture. Over the past years, numerous annotation tools have been developed in the areas of corpus linguistics and natural language processing. Among them, some have similar functionalities, e.g. Lancaster CLAWS POS tagger (Garside, 1987) and the Tree Tagger⁶ both assign POS tags to words. Some tools depend on other(s), e.g. syntactic parsers usually rely on output from POS taggers. While such a rich collection of annotation tools provides the possibility for rapid development of text mining (TM) systems by integrating existing tools, in reality we face difficulties due to their highly specialised features. Often, the major toolsets in existence have been developed in different projects at different institutes following different theoretical guidelines. As a result, many of them are incompatible with each other both in terms of annotation contents and output format.

Part of the problem derives from the annotation schemes that underpin these annotation tools. Generally, the text annotation tools are built to analyse and classify pieces of text according to certain theoretical frameworks, such as part of speech, semantic lexical taxonomy, domain-specific ontology, etc. Due to the diversity of the theories, the tools based on them tend to be highly specialised. The rules or algorithms

⁶ For further details about TreeTagger, see <http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger>.

these tools employ only work on inputs of specific content and formats, and they produce outputs of specific content in specific formats, too. Whereas such a diversity of tools would cause no problem for small-scale projects which only involve compatible tools, it can present a tough challenge for those projects which involve integration of various tools of different origins. In such circumstances, it is an important issue how to make the tools work together, i.e. the issue of tools interoperability.

In our particular case, there are two main aspects of interoperability. Firstly, it is about the sharing of analysis data and metadata. The annotation tools should be able to access the same data and “understand” the annotations produced by others. For instance, for a set of tools, such as tokeniser, POS tagger, syntactic parser, NE recogniser, etc., the rules and algorithms of the NE tool must be able to work based on the token segments, POS information and syntactic parse structure produced by the others. The other aspect concerns those tools that have the same or similar functionalities. Quite often, we need to identify the optimal tool from a set of candidates for a given task. Over decades of corpus linguistics and NLP research, multiple similar annotation tools have been developed for some annotation tasks, e.g. CLAWS, TreeTagger and Genia tagger (Tsuruoka et al., 2005) for POS annotation. With regard to interoperability, the issue here is the substitutability between them. The performances of the tools can be affected by the features and styles of text. Therefore, one may wish to quickly identify the best-performing tool(s) for a given type of text, and furthermore substitute one for another when dealing with another type of text. In reality, however, this is a goal hard to achieve because many such tools of the same functionalities take as input and produce as output different information in specific predefined formats. For example, the Enju parser developed at Tsujii Lab of Tokyo University (Miyao and Tsujii, 2005) modelled on Head-driven Phrase Structure Grammar (HPSG) (Pollard and Sag, 1994) produces predicate argument structures as the primary output while the OpenNLP syntactic parser⁷ trained on the Penn Treebank corpus produces phrase structure parse trees.

In the BOOTStrep project, we face both of these issues. This project involves annotation tools of various origins, including a term extractor developed at the National Centre for Text Mining (NaCTeM)⁸, the Genia tagger and Enju parser developed at Tsujii Lab of Tokyo University, Jena University Language and Information Engineering Lab NLP Tool Suite, a co-reference resolution tool developed at the Institute for Infocomm Research (I²R) (Yang et al., 2004a, 2004b) in Singapore, the OpenNLP toolkit, as well as other semantic analysis tools subsequently developed based on them. Among the tools involved, some are dependent on others, e.g. the syntactic parsers depend on POS taggers, and some have the same functionalities, e.g. more than one POS tagger and syntactic parser.

In order to make these tools interoperate, we need a software architecture that supports interoperability. Over the past years, much effort has been made to provide such architectures. For example, the General Architecture for Text Engineering (GATE) (Cunningham et al. 2002) provides a set of compatible annotation tools and resources within its framework, and in the SciBorg Project (Copperstake, 2006)⁹ a new mark-up language and a system have been under development aiming at combining and integrating the information produced by various annotation tools. The

⁷ See <http://opennlp.sourceforge.net/projects.html>

⁸ For the details of NaCTeM, see <http://www.nactem.ac.uk/>.

⁹ Also see <http://www.cl.cam.ac.uk/~aac10/escience/sciborg.html>

ATLAS (Laprun et al., 2002)¹⁰ Project has a similar goal. However, our observations show that UIMA provides the best solution with a more flexible and extensible architecture for implementing interoperability. Therefore, we chose UIMA for our project.

UIMA provides an architecture for sharing the analysis data and metadata among annotation tools by setting common annotation schemes, or type systems in UIMA terminology. A type system defines a set of types, e.g. Token, Sentence, Phrase, and their features, e.g. POS of a token, category of a phrase, etc. Such type systems function as the “common language” among the tools and, in order to interoperate, these tools must take as inputs and produce as outputs the types pre-defined by one or more common type systems. In practice, we do not need to develop a type system from scratch; we convert, modify and integrate the existing annotation schemes into our type systems. A type system can be either very specific for a given domain/task or generic for general linguistic annotations. In our particular case, we aim to design comprehensive type systems for bio-text mining. Therefore, we selected, modified and integrated various elements we deemed useful from a number of existing major annotation schemes.

3. Some major annotation schemes

Over the past decades, corpus and NLP research communities have dedicated remarkable efforts and time to developing annotation schemes and guidelines. Some of these annotation schemes have been well established as quasi standards in some Corpus Linguistics and NLP communities, such as the Expert Advisory Group on Language Engineering Standards (EAGLES)¹¹, the Text Encoding Initiative (TEI)¹², the Corpus Encoding Standard for XML (XCES)¹³, Penn TreeBank¹⁴, DocBook¹⁵, Dublin Core Metadata Initiative¹⁶, etc. These schemes lay out the detailed schemata to be followed for explicitly encoding information stored in natural language text in certain machine readable forms.

For example, TEI provides schema for both SGML and XML (latterly) mark-up format, and has been widely used in corpus annotation, in particular in the corpus linguistics community. Designed to cover both document macro, structural and annotation information, this schema provides a highly fine-grained schema for the marking-up and annotation of a wide range of documents in various formats. XCES is another major annotation schema which instantiates the EAGLES Corpus Encoding Standard (CES) DTDs. As its developers claim, the CES “specifies a minimal encoding level that corpora must achieve to be considered standardised in terms of descriptive representation (marking of structural and typographic information) as well as general architecture (so as to be maximally suited for use in a text database)”¹⁷.

¹⁰ Also see <http://www.nist.gov/speech/atlas/>

¹¹ See <http://xml.coverpages.org/eaglesEncod.html>

¹² See <http://www.tei-c.org/>

¹³ See <http://www.cs.vassar.edu/XCES/>

¹⁴ See <http://www.cis.upenn.edu/~treebank/>

¹⁵ See <http://www.docbook.org>

¹⁶ See <http://dublincore.org/>

¹⁷ See <http://www.cs.vassar.edu/CES/>

Today, one of the most widely accepted and applied annotation schemes in the NLP area is the Penn TreeBank schema, which provides guidelines for various annotations at POS, syntactic and semantic levels. In a recent project¹⁸, the Penn Treebank has been extended to deal with text from the Biomedicine domain.

Another annotation schema related to our work, SciXML, has been recently developed in the SciBorg Project,¹⁹ which is concerned with information extraction from Chemistry research papers (Copestake et al., 2006; Rupp et al., 2006).SciXML enables encoding and marking up of the meta information and logical structures of documents, as well as some typographical information, such as titles, authors, sections, paragraphs, tables, footnotes, italic font face, etc. It also employs a stand-off annotation model to cope with the potential problem of overlapping linguistic annotations and the XML structure trees. This is a practical schema for marking-up scientific publications, and a number of document structure types from SciXML have been adopted for our own scheme.

With respect to text mark-up in the biomedicine domain, there are several annotation schemes, including the Article DTD of BioMed Central (BMC)²⁰ and the Archiving and Interchange DTD Suite of the US National Library of Medicine (NLM)²¹. In particular, the Journal Publishing DTD²² used by the Public Library of Science (PLOS)²³, which is a subset of the Archiving and Interchange DTD Suite, provides an optimised schema for full-text journal articles. Some elements of this scheme were borrowed for our scheme.

However, most related to our work are the annotation schemes supporting a set of annotation tools which have been developed at the Tsujii Lab of the University of Tokyo²⁴, including the GENIA Corpus Annotation schema²⁵ and the Enju syntactic parse scheme²⁶. In particular, the scheme employed by the Enju Parser²⁷ provides a mechanism for annotating both the HPSG predicate-argument structures and phrase structure trees. Coupled with an efficient tool, the Enju parser, which implements a wide-coverage probabilistic HPSG grammar and an efficient parsing algorithm (Ninomiya et al., 2005) trained for the biomedicine domain (Hara et al., 2005), the Tsujii Lab schemes provide a practical solution to the bio-text processing. Their elements related to POS and syntactic parse were adopted for our scheme. Another important related work for us is the co-reference scheme and accompanying tool developed at the Institute for Infocomm Research²⁸, one of our BOOTStrep partners. This co-reference scheme has been merged into our scheme.

The annotation schemes mentioned above are but a few examples of numerous annotation schemes in existence. Our scheme is the result of adopting and modifying relevant elements from existing schemes for our own needs.

¹⁸ See <http://bioie ldc.upenn.edu/>

¹⁹ See <http://www.cl.cam.ac.uk/~aac10/escience/sciborg.html>

²⁰ See <http://www.biomedcentral.com/>

²¹ See <http://dtd.nlm.nih.gov/publishing/>

²² See <http://dtd.nlm.nih.gov/publishing/>

²³ See <http://www.plos.org/>

²⁴ See <http://www-tsujii.is.s.u-tokyo.ac.jp/>

²⁵ See <http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/>

²⁶ See <http://www-tsujii.is.s.u-tokyo.ac.jp/enju/>

²⁷ See <http://www-tsujii.is.s.u-tokyo.ac.jp/enju/>

²⁸ See <http://www.i2r.a-star.edu.sg/>

4. A UIMA annotation scheme for text mining

Aiming to annotate comprehensive information in text which is useful for text mining, our current annotation scheme encodes five major types of information, as listed below:

- 1) Meta information, such as publication date, journal, author, etc. In some cases, such information is readily available from marked-up text, such as MEDLINE abstracts.
- 2) Document structure and typographical information, such as title, paragraph, sections, bold font, etc. In some cases, such information is also readily available.
- 3) Morpho-syntactic information, such as token, lemma, POS, etc.
- 4) Syntactic structure information, such as sentence, phrase, chunk, etc.
- 5) Semantic information, such as predicate-argument relations, co-reference, biomedical entities, events, etc.

As UIMA allows a stand-off annotation model, these five annotation types form five layers of annotations, as shown in Fig. 1. In practical applications, users can select one or more layers of annotation depending on their specific requirements. In some cases, where multiple tools of the same functionality produce overlapping annotations, their annotation outputs can be stored as extra annotation sub-layers. For example, if two syntactic parsers classify the same phrase into different categories, or assign the same category to overlapping text spans, their outputs can be stored separately in two sub-layers of syntactic parsing. Below we explain further the five layers of annotation we have specified.

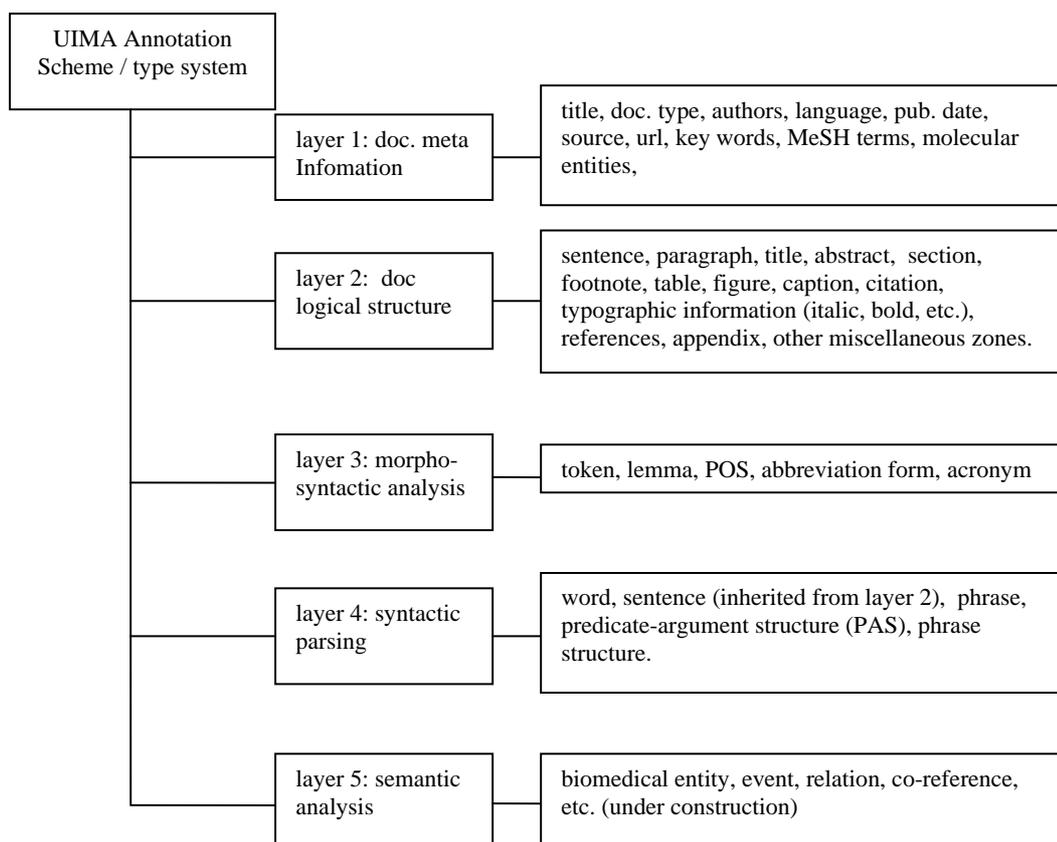


Figure 1: Outline of the UIMA annotation scheme/type system.

3.1 Document meta information

This layer of annotation contains the types and features describing a document as a whole. The main type is named *Meta*, which contains a set of attributes whose values describe different aspects of the document under consideration. These attributes include *docType* (abstract, research article, review, etc.), *source* (the origin of the document), *title*, *authors*, *MeSHTerms* (Medical Subject Headings), *molecularEntities* (Gene entities occurring in the document), etc. As shown, the majority of the attributes describe the general aspects of a document, but some are designed to cater for the specific requirement of bio-text mining applications. A typical feature in this regard is MeSH, which is a controlled vocabulary thesaurus compiled by the US National Library of Medicine (NLM)²⁹. It is used for indexing articles from biomedical journals for the MEDLINE/PubMED database by NLM, and it can potentially be applied to various other tasks.

It is a challenging task to automatically extract all the meta information about documents. Fortunately, there exists a substantial amount of bio-texts available from open access publishers that are marked up with much of the meta information. In such cases, it is possible to automatically map the relevant type of information to our scheme. Due to the difficulty of automatic extraction of such information from raw texts, most of the attributes are set as optional.

²⁹ For further details about MeSH, see <http://www.nlm.nih.gov/mesh/>.

3.2 Document logical structure annotation

Document logical structural information forms another layer of the annotation. The document structure annotation centres on *Zone* type. A *Zone* refers to a distinct division of text. It is an abstract neutral type which provides a parent type for various sub-types related to document structure, such as *Sentence*, *Paragraph*, *Section*, etc. We also included typographic information, e.g. bold, italic, underscore, etc., in this layer of annotation.

The advantage of having structural information specified for text mining is that we can select those text units/segments in which we are interested. In a document, different text zones can have different roles and serve different purposes. Take an typical academic paper, for example: the title represents the general theme of the paper, the abstract provides a general insight into the contents of the paper, the survey section discusses related work carried out in the past, one or more core sections present the author's own work (methodology, results, evaluation, discussion, etc.), the conclusion section makes a summary of what has been presented, the acknowledgement section acknowledges funders and support the author received, and so on.

With respect to text mining applications, we may need to identify different types of information from different sections of the documents. For example, if we want to identify the work done by the author, we would mainly need to collect the sections presenting the author's own work. The unrelated sections, such as survey and acknowledgement, would only cause "noise". In contrast, if one is interested to know which funders support which type of work, the acknowledgement sections would become critical.

In order to cater for such requirements of text mining, we defined general types of text zones occurring in academic publications. Currently, our scheme encapsulates the text zones with two main types: *textBody* and *miscPart* (miscellaneous part). Each of these contains a set of attributes which in turn are references to other sub-zone types. For example, *textBody* contains the attributes of *title*, *authorInfoList*, *abstract*, *sectionList*, *footnoteList*, *citationList*, *appendixList* while *miscPart* contains the attributes of *tableList*, *figureList* and *miscList* (list of any miscellaneous zones that are hard to pre-define).

Again, much of such information is often available from the mark-up produced by publishers of academic journals. Nonetheless, it is a difficult task to extract all such information from plain text files, and efficient tools for this purpose are yet to be developed.

3.3 Morpho-syntactic and typographical annotation

This layer of the annotation scheme contains the types and features related to the morpho-syntactic information regarding words and tokens, such as tokenization, lemmatization, short form, etc. The main type here is *Token*, which contains attributes *POS*, *lemma*, *shortForm*, *orthographic*, etc. Of these attributes, *shortForm* is a reference to another type *ShortForm*, which in turn has two sub-types *Abbrev* (abbreviation) and *Acronym*.

Usually, the short forms of tokens/terms may be only a marginal issue in general corpus linguistics, but for bio-text mining their identification is a critical

issue. In bio-text mining, it is an essential task to identify and map acronyms and their full forms, as academic papers contain a large amount of such terms (Okazaki and Ananiadou; 2006)³⁰.

The *POS* attribute takes values from the Genia POS tagset (Kim et al., 2003), which is a slight modification of the Penn TreeBank POS tagset. The main reason for this is that we use the Genia POS tagger, which is trained on the Genia corpus. In effect, the PennTree bank and Genia tagsets can be easily mapped to each other.

We include the *orthographic* attribute for capturing some special features of tokens which may serve as anchors for important pieces of information. For instance, it has a value of “*capitalised*”, which can be useful for detecting proper nouns. Another example is “*alphanumeric*” for tokens like *p33* which can provide cues for finding information related to page numbers.

3.4 Syntactic parsing

This part of the annotation scheme defines the types and attributes pertaining to the syntactic structure of sentences, including both chunking and full parsing. A problem we faced here is that the variety of syntax theories has led to various parsing tools producing different parse outputs, particularly between the Enju parser and the current OpenNLP parser included in our project (refer to section 2). We found it is difficult to map either of their outputs to the other, and consequently we decided to have parallel syntactic annotation sub-layers in the scheme. In this paper, we focus on the Enju parse annotation scheme.

The Enju parse scheme is based on HSPG, and thus focuses on the predicate-argument relations between words. As a result, the main type in this layer is *Word* which contains attributes pointing to arguments/modifiers. Each *Word* type has up to five arguments or a modifier. In addition, the attributes encode other information including POS, syntactic functional categories (noun phrase, verb phrase, etc.) and the parent syntactic constituent of a given word. The Enju annotation scheme also allows secondary phrase structure representation, in which the main type *Phrase* has attributes for encoding information including syntactic category, syntactic head word, semantic head word, etc. This annotation layer provides critical information for advanced text mining applications which involve analysis of complex relations between syntactic constituents.

3.5 Semantic annotation

The semantic layer of the annotation aims to provide a scheme for encoding co-reference relations and biology domain specific concepts and events. For the co-reference annotation, we adopted the scheme developed at the Institute for Infocomm Research in Singapore. With regard to domain specific semantic information, as the BOOTStrep project focuses on the biological subdomain of Gene Regulation, this layer of annotation is targeted at those entities, events and relations occurring in this domain. It is still under construction and improvement, and we will draw upon relevant biological ontologies.

³⁰ Also see website: <http://www.nactem.ac.uk/software/acromine>.

Our annotation scheme is under further improvement and modification to better meet the needs of bio-text mining applications. Furthermore, tools are yet to be developed for populating all the information prescribed by our scheme. However, the core parts of the scheme, including some parts of document structure, POS annotation, syntactic parsing, named entities, co-reference, etc. are already supported by working software.

4. Summary

We presented an annotation scheme which is designed for bio-text mining and provides a type system for tool interoperability within UIMA. Our scheme is based on a number of existing established schemes, particularly those coupled with effective working annotation tools. In addition, our scheme contains elements pertaining to the specific requirements of bio-text mining. Nonetheless, the major part of the scheme is compatible with generic corpus annotation tasks, and many tools involved in our work are portable to other domains or general corpus linguistics study. Our work offers valuable experience for the text mining, NLP and corpus linguistics communities in developing large-scale corpus annotation software systems based on existing schemes and tools.

Acknowledgements

Our work is supported by the EC BOOTStrep Project (Ref. FP6 – 028099). The UK National Centre for Text Mining is supported by the JISC, BBSRC and EPSRC.

References

- Ananiadou, S. and J. McNaught. (2006). Text Mining for Biology and Biomedicine. Norwood, MA: Artech House.
- Copestake, A., P. Corbett, P. Murray-Rust, CJ Rupp, A.Siddharthan, S. Teufel and B. Waldron. (2006). An architecture for language processing for scientific texts. *Proceedings of the UK e-Science All Hands Meeting 2006*. Nottingham, UK.
- Cunningham, H., D. Maynard, K. Bontcheva, V. Tablan. (2002). GATE: A framework and graphical development environment for robust NLP tools and applications. *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*. Philadelphia, July 2002.
- Ferrucci, D. and A. Lally. (2004). Building an example application with the Unstructured Information Management Architecture, *IBM Systems Journal* 43, No. 3, 455–75.
- Garside, R. (1987). The CLAWS word-tagging system. In: R. Garside, G. Leech and G. Sampson eds), *The Computational Analysis of English: A Corpus-based Approach*. London: Longman.
- Hara, T., Y. Miyao and J. Tsujii. (2005). Adapting a probabilistic disambiguation model of an HPSG parser to a new domain. In Robert Dale, Kam-Fai Wong, Jian Su and Oi Yee Kwong (Eds.), *Natural Language Processing – IJCNLP*

2005. *Lecture Notes in Artificial Intelligence* 3651. Jeju Island, Korea, pp. 199-210, Springer-Verlag, ISSN 0302-9743.
- Kim, J.D., T. Ohta, Y. Teteisi and J. Tsujii. (2003). GENIA corpus - a semantically annotated corpus for bio-textmining. *Bioinformatics* 19(suppl. 1). pp. i180-i182.
- Laprun, C., J.G. Fiscus, J. Garofolo, and S. Pajot (2002). A practical introduction to ATLAS. *Proceedings LREC 2002*, Las Palmas.
- Miyao, Y. and J. Tsujii (2005). Probabilistic disambiguation models for wide-coverage HPSG parsing. In *Proceedings of ACL-2005*, pp. 83-90.
- Ninomiya, T., Y. Tsuruoka, Y. Miyao and J. Tsujii (2005). Efficacy of beam thresholding, unification filtering and hybrid parsing in probabilistic HPSG parsing. In *Proceedings of the 9th International Workshop on Parsing Technologies (IWPT 2005)*.
- Okazaki, N., S. Ananiadou. (2006). Building an abbreviation dictionary using a term recognition approach. *Bioinformatics*, 22(24), pp. 3089-95.
- Pollard, C. and I. Sag. (1994). *Head-Driven Phrase Structure Grammar*. Chicago: University of Chicago Press.
- Rupp, C.J., A. Copestake, S. Teufel, B. Waldon. (2006). Flexible interfaces in the application of language technology to an eScience corpus. *Proceedings of the UK e-Science All Hands Meeting 2006*. Nottingham, UK.
- Tsuruoka, Y., Y. Tateishi, J.D. Kim, T. Ohta, J. McNaught, S. Ananiadou and J. Tsujii. (2005). Developing a robust part-of-speech tagger for biomedical text, *Advances in Informatics - 10th Panhellenic Conference on Informatics*, LNCS 3746, pp. 382-92.
- Yang, X., J. Su, G. Zhou and C. L. Tan. (2004a). An NP-Cluster approach to coreference resolution, In *Proceedings of the 20th International Conference on Computational Linguistics (COLING04)*, Geneva, Switzerland.
- Yang, X., J. Su, G. Zhou and C. L. Tan. (2004b). Improving pronoun resolution by incorporating coreferential information of candidates. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL04)*, Barcelona, Spain. pp. 127-34.