# A comparative evaluation of modern English corpus grammatical annotation schemes

***Eric Atwell, George Demetriou, John Hughes,***
***Amanda Schiffrin, Clive Souter and Sean Wilcock***
***Centre for Computer Analysis of Language and Speech (CCALAS)***

## *1 Introduction*

Many English Corpus Linguistics projects reported in *ICAME Journal* and elsewhere involve grammatical analysis or tagging of English texts (eg Atwell 1983, Leech et al 1983, Booth 1985, Owen 1987, Souter 1989a, O'Donoghue 1991, Belmore 1991, Kytö and Voutilainen 1995, Aarts 1996, Qiao and Huang 1998). Each new project has to review existing tagging schemes, and decide which to adopt and/or adapt. The AMALGAM project can help in this decision, by providing descriptions and analyses of a range of tagging schemes, and an internet-based service for researchers to try out the range of tagging schemes on their own data.

The project AMALGAM (Automatic Mapping Among Lexico-Grammatical Annotation Models) explored a range of Part-of-Speech tagsets and phrase structure parsing schemes used in modern English corpus-based research. The PoS-tagging schemes include: Brown (Greene and Rubin 1981), LOB (Atwell 1982, Johansson et al 1986), Parts (man 1986), SEC (Taylor and Knowles 1988), POW (Souter 1989b), UPenn (Santorini 1990), LLC (Eeg-Olofsson 1991), ICE (Greenbaum 1993), and BNC (Garside 1996). The parsing schemes include some which have been used for hand annotation of corpora or manual post-editing of automatic parsers, and others which are unedited output of a parsing program. Project deliverables include:

– a detailed description of each PoS-tagging scheme, at a comparable level of detail. This includes a list of PoS-tags with descriptions and example uses from the source Corpus. The description of the use of PoS-tags is also illustrated in a multi-tagged corpus: a set of sample texts PoS-tagged in parallel with each PoS-tagset (and proofread by experts), for comparative studies

– an analysis of the different lexical tokenization rules used in the source Corpora, to arrive at a 'Corpus-neutral' tokenization scheme (and consequent adjustments to the PoS-tagsets in our study to accept modified tokenization)
– an implementation of each PoS-tagset in conjunction with our standardised tokenizer, as a family of PoS-taggers, one for each PoS-tagset
– a method for 'PoS-tagset conversion', taking a text tagged according to one PoS-tagset and outputting the text annotated with another PoS-tagset
– a sample of texts parsed according to a range a parsing schemes: a Multi-Treebank resource for comparative studies
– an Internet service allowing researchers worldwide free access to the above resources, including a simple email-based method for PoS-tagging any English text with any or all PoS-tagset(s).

## *2 Defining the PoS-tagging schemes*

Grammatical analysis is usually divided into two levels or phases:

1. Lexico-grammatical wordclass annotation, also called morphosyntactic annotation, or Part-of-Speech wordtagging, or just 'tagging'
2. Syntactic structure annotation, or full parsing

Even at the first level, there is great diversity of annotation schemes or models. Here an example sentence is tagged according to several alternative tagging schemes and vertically aligned:

| | Brown | ICE | LLC | LOB | PARTS | POW | SEC | UPenn |
|---|---|---|---|---|---|---|---|---|
| select | VB | V (montr, imp) | VA+0 | VB | adj | M | VB | VB |
| the | AT | ART (dof) | TA | ATI | art | DD | ATI | DT |
| text | NN | N (com, sing) | NC | NN | noun | H | NN | NN |
| you | PPSS | PRON (pers) | RC | PP2 | pron | HP | PP2 | PRP |
| want | VB | V (montr, pres) | VA+0 | VB | verb | M | VB | VBP |
| to | TO | PRTCL (to) | PD | TO | verb | I | TO | TO |
| protect | VB | V (montr, infin) | VA+0 | VB | verb | M | VB | VB |
| . | . | PUNC (per) | . | . | . | . | . | . |

Note the differences in representation, more crucially, in *delicacy* or level of detail in grammatical classification. Delicacy is a factor in evaluation: a delicate analysis is more difficult, so a 'skeletal' PoS-tagger or parser will score higher

accuracy ratings. An indelicate annotation is sufficient for many NLP applications, eg grammatical error detection in Word Processed text (Atwell 1983), training Neural Networks (Benello et al 1989), or training statistical language processing models (Manning and Schutze 1999).

A problem in making a comparison between schemes is in identifying where a scheme is precisely defined. Should we consider the definitive description of a scheme to be

1. The annotation description/handbook (if it exists), or
2. The proof-read annotated corpus, which we can use to 'train' our definitive model, or
3. The expert intuitions of the linguist in charge of the corpus annotation project?

Obviously each of these is fallible, in that it may contain errors (in the case of the corpus) omissions (in the case of the handbook) or make mistakes (in the case of the linguist). Furthermore, handbook and/or expert linguist may be unavailable for one or more schemes under consideration. In some ways this is reassuring, because it means we can (and must) accept a tagger accuracy rate of less than 100 per cent.

It is relatively easy to acquire a list of tags in a given tagset, extracting this from the Corpus or the manual. However, even this raises some problems. Some of the schemes we chose did not actually contain tags for the written texts we want to be able to process. POW, for example, has no tags for punctuation, since it is a spoken corpus. These tags have had to be created. It also lacks tags for numbers, dates etc.

The definition of each tag was even more problematic. As Corpus Linguists, we preferred to see the tagged corpus as definitive of the meanings and uses of tags in a tagset. However, the training corpora were not equal in size, genre, etc. In some cases, examples of word and tag co-occurrences in the training corpus are so rare that one cannot be certain they have been correctly tagged. Consequently the tagger's performance in this area is likely to be idiosyncratic. This also means that evidence may be too scarce to help decide whether or not an error has in fact been made by the tagger. The same issue arises in the identification of words. Even in the small amount of text we chose as test material (see section 4), two items occurred (*pabs, cart's* in COLT) which are difficult to classify.

Capitalised words are problematic since the training corpus alone cannot always tell us how to distinguish proper nouns from words capitalised at the start of sentences, headings, etc. There are cases where more than one tag may be legitimate; for example, LOB has a tag for foreign words, so should *Buenos Aires* be tagged as a proper noun or foreign word?

We have compiled a detailed description of each PoS-tagging scheme, at a comparable level of detail for each Corpus annotation scheme: a list of PoS-tags with descriptions and example uses from the source Corpus. For closed class categories (eg preposition), this includes a full list of words with that tag in the training corpus; for open class categories (eg common noun), a list of examples from the training corpus is cited.

We have also compiled a multi-tagged corpus, a set of sample texts PoS-tagged in parallel with each PoS-tagset (and proofread by experts), to use as a benchmark against which other taggers/parsers could be measured, and for linguistic comparison of schemes (size, delicacy, notational/theoretical differences). We selected material from three quite different genres of English: informal speech of London teenagers from COLT, the Corpus of London Teenager English (Andersen and Stenström 1996); prepared speech for radio broadcasts from SEC, the Spoken English Corpus (Taylor and Knowles 1988); and written text in software manuals from IPSM, the Industrial Parsing of Software Manuals corpus (Sutcliffe et al 1996). Sixty sentences were chosen from each:

| | **Sentences** | **Words** |
|---|---|---|
| **London teenager speech (COLT)** | 60 | 407 |
| **Radio broadcasts (SEC)** | 60 | 2016 |
| **Software manuals (IPSM)** | 60 | 1016 |
| **Total:** | 180 | 3439 |

The IPSM software manual data was chosen because it had already been used at an evaluation workshop for parsing programs (see section 6). Each of the sentences was tagged using the multi-tagger trained for the schemes used in Brown, ICE, LLC, LOB, Unix Parts, POW, SEC and UPenn (see section 4). BNC tagging was kindly provided by Lancaster University, using the CLAWS C5 and C6 tagsets. The outputs of the taggers were then proof-read and post-edited by experts in each scheme.

## *3 A neutral tokenization scheme*

Preparing the original corpora (ICE, POW, SEC etc) for training by a 'tagset-learning' tagger such as Brill's is non-trivial. For an indication of the work involved, first consider the Brown corpus. It is all in upper case, so a program is required to convert the format of Brown to lower case. Next, Brown uses 'combined' tags for words like *won't* whereas most other corpora split combined words up into constituent parts. For consistency Brown needs to have combined words and their associated tags split into constituent parts. Most tagged corpora are formatted vertically with one word per line. On each line there would typically be the word, the tag for the word and some reference information. In contrast, the Brill tagger takes input in horizontal format so the file needs to be reformatted horizontally before training can take place.

A further problem for us was corpus texts which were transcriptions of spoken dialogue, such as the London-Lund Corpus. For example, the learning algorithm generally makes use of punctuation to guide analysis, and can be misled by hesitations, false starts etc. The main issue is that the London-Lund and POW schemes are designed for a transcribed spoken corpus, whereas we want to be able to apply the scheme to written corpora. So, spoken text had to be pre-processed to add some punctuation, and remove markup for pauses, repeated phrases, inaudible text etc. Our editing process is analogous to the editing of Hansard, the official transcripts of parliamentary debates. Canadian Hansard transcripts have been widely used in NLP research, as 'well-behaved' spoken text.

Different word-tagging schemes assume different segmentation of text into lexical units or words, and have different ways of handling punctuation. For example, sometimes compound names or idiomatic phrases are given a single wordtag; in contrast, sometimes affixes are stripped off and given a separate word-tag. It was therefore necessary for the alignment program, which displays rival taggings of the same text, to be quite sophisticated. The segmentation of text into lexical items is often called tokenisation. Standard rules for tokenisation were needed not only to intelligently align the alternative tags for each scheme, but more importantly to pre-segment text before it is presented to the tagger. A detailed description of the tokenisation process is given on our website. The performance of the tagger could be improved by incorporating bespoke tokenisers for each scheme, but we have compromised by using only one for all schemes, to simplify comparisons. This results in errors of the following kind, using examples from the POW scheme:

| | **Tokeniser/ Tagger output** | **Correct analysis in POW corpus** |
|---|---|---|
| ***Negatives*** | *are*/OM n't/OXN | *aren't*/OMN |
| ***Enclitics*** | *where's*/H | *where*/AXWH *'s*/OM |
| ***Possessives*** | *God's*/HN | *God*/HN *'s*/G |
| ***Expressions*** | *for*/P example/H<br>*have*/M to/I | *for-example*/A<br>*have-to*/X |
| | (similarly for *set-up, as-well-as, so-that, next-to, Edit/Copy, Drag & Drop, Options...* etc.) | |

## *4 The AMALGAM multi-tagger: a family of PoS-taggers*

The tagging approach we eventually adopted does not involve a hand-crafted set of tagging or mapping rules. Although some of the tagging programs used to annotate the LOB, ICE, SEC, etc corpora were available to us and use similar underlying algorithms, they differ in significant ways. In some cases we do not have access to the tagging programs, and in one case (POW) the corpus was tagged 'manually' by linguists. We decided to train a publicly-available machine learning system, the Brill tagger (Brill, 1993), to re-tag according to all of the schemes we are working with. As the Brill tagger was the sole automatic annotator for the project, we achieved greater consistency.

The Brill system is first given a tagged corpus as a training set, to extract or learn a complex set of tagging rules for the given lexico-grammatical annotation model. Then, the learnt rules can be applied to a new text, to annotate with the given tagset. We accept the original tagged/parsed corpus itself as definitive of the tagging/parsing scheme for that corpus.

The procedure for training Brill's tagger with a new scheme, once the input corpus has been pre-formatted, is described in detail on our website. A lexicon is extracted from the training corpus, and two sets of non-stochastic rules are derived. The lexicon contains each word type found in the corpus, along with a frequency ordered list of the tags with which the word has been labelled. The first set of rules is *contextual*, indicating which tag should be chosen in the context of other tags or words. The rules are generated from a small set of templates, and then refined iteratively to minimise the error rate. The second set is *lexical*, and is used to guess the tag for words which are not found in the lexicon. Essentially, they contain morphological information, which indicates word

class. Because the rules are refined iteratively, the training process can last hours, or even days, but only needs to be conducted once!

The model learnt encapsulates the tagging scheme applied in the training corpus. An indication of the relative success of this method of 'learning' a tagging scheme can be gleaned when the tagger is run on 10,000 words of training text, as shown in Table 1:

*Table 1:* Model size and accuracy of the re-trained Brill multi-tagger

| Tagger | Lexicon | Context Rules | Lexical Rules | Accuracy % |
|---|---|---|---|---|
| Brown | 53113 | 215 | 141 | 97.43 |
| ICE | 8305 | 339 | 128 | 90.59 |
| LLC | 4772 | 253 | 139 | 93.99 |
| LOB | 50382 | 220 | 94 | 95.55 |
| Unix Parts | 2842 | 36 | 93 | 95.8 |
| POW | 3828 | 170 | 109 | 93.44 |
| SEC | 8226 | 206 | 141 | 96.16 |
| UPenn | 93701 | 284 | 148 | 97.2 |

The error rate varies from 2.57 per cent for Brown to 9.41 per cent, as a result of differences in the size of the available training material, and in the delicacy of the scheme. The most common errors for six of the schemes, (as a percentage of all errors for that scheme), are as follows:

*Brown* VBN/VBD 14.6% JJ/NN 4.9% NN/VB 4.2%

*ICE* V(cop,pres,encl)/V(intr,pres,encl) 4.1% ADJ/N(prop,sing) 3.1% PUNC(oquo)/PUNC(cquo) 2.6%

*LLC* PA/AC 4.1% PA/AP 2.7% RD/CD 2.7%

*LOB* IN/CS 5.8% TO/IN 4.1% VBN/VBD 4%

*POW* AX/P 4.3% OX/OM 2.9% P/AX 2.5%

*SEC* TO/IN 6.3% JJ/RB 5.6% JJ/VB 4.8%

A more realistic evaluation of tagger accuracy across a range of text types was derived in building the multi-tagged corpus described in Section 2. Samples of teenage conversations, radio scripts, and software manuals were tagged by the multi-tagger. The outputs of the multi-tagger were then proof-read and post-edited by experts in each scheme. Table 2 shows the accuracy of each tagger for the multi-tagged corpus. All the tagging schemes performed significantly worse on this test material than they did on their training material, which indicates how non-generic they are. With the exception of POW, the schemes all performed worst on the COLT data, which is informal speech. The POW scheme, which was designed for informal spoken data happily performs better on the COLT material than on prepared speech or written software manuals. Nevertheless it was outperformed on the informal spoken material by the UPenn scheme, because this is a less delicate scheme.

*Table 2:* Accuracy found in manual proof-reading of multi-tagged corpus:

| TAGSET | TOTAL | IPSM60 | COLT60 | SEC60 |
|---|---|---|---|---|
| Brown | 94.3 | 94.3 | 87.7 | 95.6 |
| UPenn | 93.1 | 91.6 | 88.7 | 94.6 |
| ICE | 89.6 | 87.0 | 85.3 | 91.8 |
| Parts (Unix) | 86.7 | 89.9 | 82.3 | 86.0 |
| LLC | 86.6 | 86.9 | 84.3 | 87.0 |
| POW | 86.4 | 87.6 | 87.7 | 85.4 |

These results indicate that there is still work to be done in creating a generic tagging scheme which performs equally well regardless of the text genre.

## *5 Mapping between tagging schemes*

The tag-mapping approach we eventually adopted does not involve a hand-crafted set of mapping rules. Instead, to re-tag the old parts of speech of a corpus with a new scheme of another, we apply our tagger to just the words of the corpus. For example, mapping from SEC to Brown lexico-grammatical annotation models is achieved by re-tagging SEC text with Brown tags:

1. learn n-gram-like tagger rules from Tagged Brown Corpus
2. apply rule-based tagger learnt from Brown to SEC text

This might appear to be 'cheating', but we believe this approach is more appropriate. Earlier experiments with devising a set of mapping rules from one tagset to another (Hughes and Atwell 1994, Atwell et al 1994, Hughes et al 1995) concluded that one-to-many and many-to-many mappings predominated over simple one-to-one (and many-to-one) mappings, resulting in more errors than the apparently naïve approach of ignoring the source tags.

## *6 A MultiTreebank*

The differences between English corpus annotation schemes are much greater between parsing schemes for full syntactic structure annotation than they are at word class level. The following are parses of the sentence '*Select the text you want to protect.'* according to the parsing schemes of several English parsed corpora or treebanks:

==> ENGCG-BankOfEnglish <==
"select" <*>  V IMP VFIN  @+FMAINV
"the"  DET CENTRAL ART SG/PL  @DN>
"text" N NOM SG @OBJ
"you"  PRON PERS NOM SG2/PL2 @SUBJ
"want"  V PRES -SG3 VFIN  @+FMAINV
"to" INFMARK>  @INFMARK>
"protect"  V INF @-FMAINV
"<$.>"

==> L-ATR <==
[start [sprpd1 [sprime1 [sc3 [v2 Select_VVIOPTATIVE [nbarq14 [nbar4 [d1 the_AT d1] [n1a text_NN1MESSAGE n1a] nbar4] [sd1 [nbar6 you_PPY nbar6] [vbar1 [v33 want_VVOMENTAL-ACT [vibar1 to_TO [v1 protect_VVIHELP v1] vibar1] v33] vbar1] sd1] nbarq14] v2] sc3] sprime1] ._. sprpd1] start]

==> LOB-TREEBANK <==
[S[V[VB[ select ]VB]V][Ns[ATI[ the ]ATI][NN[ text ]NN][Fr[N[PP2[ you ]PP2]N][V[VB[ want ]VB]V][Ti[Vi[TO[ to ]TO][VB[ protect ]VB]Vi]Ti]Fr]Ns][.[ . ].]S]

==> POW <==
Z CL 1 M SELECT 1 C NGP 2 DD THE 2 H TEXT 2 Q CL 3 S NGP HP YOU 3 M WANT 3 C CL 4 I TO 4 M PROTECT 1 ? .

There is even greater diversity in the parsing schemes used in alternative NLP parsing *programs*. The example sentence was actually selected from a test-set used at the Industrial Parsing of Software Manuals workshop (Sutcliffe et al 1996); it is one of the shortest test sentences, which one might presume to be one of the most grammatically straightforward and uncontroversial.

In order to get a better picture of the differences between parsing schemes (or perhaps to muddy the waters still further), Atwell (1996) proposed the collection of a small multi-parsed corpus. 60 of the English sentences from software manuals included in our multi-tagged corpus were selected. These had the benefit of already having been automatically parsed using a number of different parsing programs, so comparison would be possible with the schemes used by linguists in the hand annotation of corpora.

The parsing schemes exemplified in our MultiTreebank include some which have been used for hand annotation of corpora or manual post-editing of automatic parsers: EPOW (O'Donoghue 1991), ICE (Greenbaum 1992), POW (Souter 1989a,b), SEC (Taylor and Knowles 1988), and UPenn (Marcus et al 1993). Linguist experts in each of these corpus annotation schemes kindly provided us with their parsings of the 60 IPSM sentences. Others are unedited output of parsing programs: Alice (Black and Neal 1996), Carroll/Briscoe Shallow Parser (Briscoe and Carroll 1993), DESPAR (Ting and Shiuan 1996), ENGCG (Karlsson et al 1995; Voutilainen and Järvinen 1996), Grammatik (WordPerfect 1998), Link (Sleator and Temperley 1991; Sutcliffe and McElligott 1996), PRINCIPAR (Lin 1994, 1996), RANLP (Osborne 1996), SEXTANT (Grefenstette 1996), and TOSCA (Aarts et al 1996; Oostdijk 1996). Language Engineering researchers working with these systems kindly provided us with their parsings of the 60 IPSM sentences.

The MultiTreebank illustrates the diversity of parsing schemes available for modern English language corpus annotation. The most significant differences are:

- Varying theoretical backgrounds: eg dependency vs phrase structure
- Genre-specific tokenization, eg of punctuation, keyboard characters
- Genre-specific focus of grammar, eg spoken dialogue vs written text
- Parser developers focus on applications, eg information extraction, rather than full linguistic annotation
- Parser output format may be computer oriented rather than easily proof-readable
- Parser output was not hand edited, but best solution was chosen
- Parser output offers reader few precedents, whereas a corpus may offer more.

The EAGLES project (1996) has embarked upon the task of setting standards for corpus annotation. Its guidelines recognise layers of syntactic annotation, which form a hierarchy of importance. None of the parsing schemes included here contains all the layers (*a-h*, in Table 3 below). Different parsers annotate with different subsets of the hierarchy.

*Table 3*: Evaluation of IPSM Grammatical Annotation Models, in terms of EAGLES layers of syntactic annotation

(a) Bracketing of segments
(b) Labelling of segments
(c) Showing dependency relations
(d) Indicating functional labels
(e) Marking sub-classification of syntactic segments
(f) Deep or 'logical' information
(g) Information about the rank of a syntactic unit
(h) Special syntactic characteristics of spoken language

| Parse Scheme | EAGLES layer | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | a | b | c | d | e | f | g | h | Score |
| ALICE | yes | yes | no | no | no | no | no | no | 2 |
| CARROLL | yes | yes | no | no | no | no | no | no | 2 |
| DESPAR | no | no | yes | no | no | no | no | no | 1 |
| ENGCG | no | no | yes | yes | yes | no | no | no | 3 |
| EPOW | yes | yes | no | yes | no | no | no | yes | 4 |
| GRAMMATIK | yes | yes | no | yes | no | no | no | no | 3 |
| ICE | yes | yes | no | yes | yes | no | no | yes | 5 |
| LINK | no | no | yes | yes | no | no | no | no | 2 |
| POW | yes | yes | no | yes | no | yes | no | yes | 5 |
| PRINCIPAR | yes | yes | yes | no | no | yes | yes | no | 5 |
| RANLT | yes | yes | no | no | no | yes | yes | no | 4 |
| SEC | yes | yes | no | no | yes | no | no | yes | 4 |
| SEXTANT | yes | yes | yes | yes | no | no | no | no | 4 |
| TOSCA | yes | yes | no | yes | yes | yes | no | yes | 6 |
| UPENN | yes | yes | no | no | no | no | no | no | 2 |

Each cell in the table is labelled **yes** or **no** to indicate whether a MultiTreebank parsing scheme includes an EAGLES layer (at least partially). The **score** is an indication of how many layers a parser covers.

The rather disheartening conclusion we can draw from these observations is that it is difficult, if not impossible, to map between all the schemes. Unlike the tagging schemes, it does not make sense to make an application-independent comparative evaluation. No single standard can be applied to all parsing projects. Even the presumed lowest common denominator, bracketing, is rejected by some corpus linguists and dependency grammarians. The guiding factor in what is included in a parsing scheme appears to be the author's theoretical persuasion or the application they have in mind.

## *7 Website and e-mail tagging service*

The multi-tagged corpus, MultiTreebank, tagging scheme definitions and other documentation are available on our website. The multi-tagger can be accessed via email: email your English text to *amalgam-tagger@scs.leeds.ac.uk*, and it will be automatically processed by the multi-tagger, and then the output is mailed back to you. The tagger is intended for English text; it will not work correctly for languages other than English (some users have tried!). The text must be the main body of the text, NOT an attachment, and must be raw text, NOT a Word.doc or other format. By sending a blank message to *amalgam-tagger@scs.leeds.ac.uk* with *help* as the subject you will receive a help file instructing you how to use the multi-tagger. The text to be tagged can be first passed through the tokeniser, which applies various formatting rules to the text. This can be turned off and on by setting a flag in the subject line of your mail. Users can select any or all of the eight schemes (Brown, ICE, LLC,LOB, Parts, POW, SEC, UPenn). The tagged text is returned one email reply message per scheme. A verbose mode can also be selected, which gives the long name for each tag as well as its short form in the output file.

The service has been running since December 1996, and usage is logged on our website; up to December 1999, it processed 19,839 email messages containing over 628 megabytes of text. The most popular schemes are LOB, UPenn, Brown, ICE, and SEC (in that order), with relatively little demand for Parts, LLC, and POW; this reflects the popularity of the source corpora in the Corpus Linguistics community. Apart from obvious uses in linguistic analysis, some unforeseen applications have been found, eg in using the tags to aid data compression of English text (Teahan 1998) and as a possible guide in the search for extra-terrestrial intelligence (Elliott and Atwell 2000).

## *8 Conclusions*

NLP researchers have not agreed on a standard lexico-grammatical annotation model for English. As there is no single standard, the AMALGAM project has investigated the range of alternative schemes. We have trained a 'machine learning' tagger with several lexico-grammatical annotation models, to enable it to annotate according to several rival modern English languge corpus Part-of-Speech tagging schemes. To map from one tagging scheme to another, we first strip the 'source' tags, and re-tag the text with "target" tags. Our main achievements are:

**Software**: *PoS-taggers* trained to annotate text according to several rival lexico-grammatical annotation models, accessible over the Internet via email.

**Data-sets**: a *multi-tagged corpus* and *multi-treebank*, a corpus of English text where each sentence is annotated according to several rival lexico-grammatical annotation models. We have also collected together definitions of eight major English corpus word-tagging schemes, as a resource for comparative study.

**Knowledge**: a clearer understanding of the differences and similarities between rival lexico-grammatical annotation models. Our results for tagger accuracy on texts of different genres indicate that there is still work to be done on the creation of a truly generic tagging scheme.

## *Acknowledgements*

We are grateful to the UK Engineering and Physical Sciences Research Council (EPSRC) for funding this research project. We are also indebted to the numerous researchers worldwide who helped us by providing advice, data, and documentation, and by proofreading the multi-tagged Corpus and MultiTreebank.

## *References*

Aarts, Jan. 1996. A tribute to W. Nelson Francis and Henry Kucera: grammatical annotation. *ICAME Journal* 20:104–107.

Aarts, Jan, Hans van Halteren and Nelleke Oostdijk. 1996. The TOSCA analysis system. In C. Koster and E. Oltmans (eds). *Proceedings of the first AGFL workshop.* 181–191. Technical Report CSI-R9604, Computing Science Institute, University of Nijmegen.

Andersen, Gisle, and Anna-Brita Stenström. 1996. COLT: a progress report. *ICAME Journal* 20:133–136.

Atwell, Eric. 1982. *LOB Corpus tagging project: post-edit handbook*. Department of Linguistics and Modern English Language, University of Lancaster.

Atwell, Eric. 1983. Constituent likelihood grammar. *ICAME Journal* 7:34–66.

Atwell, Eric. 1996. Comparative evaluation of grammatical annotation models. In R. Sutcliffe et al (1997), 25–46.

Atwell, Eric, John Hughes and Clive Souter. 1994. AMALGAM: Automatic Mapping Among Lexico-Grammatical Annotation Models. In J. Klavans and P. Resnik (eds.), *The balancing act – combining symbolic and statistical approaches to language. Proceedings of the workshop in conjunction with the 32nd annual meeting of the Association for Computational Linguistics.* New Mexico State University, Las Cruces, New Mexico, USA.

Belmore, Nancy. 1991. Tagging Brown with the LOB tagging suite. *ICAME Journal* 15:63–86.

Benello, Julian, Andrew Mackie and James Anderson. 1989. Syntactic category disambiguation with neural networks. *Computer Speech and Language* 3:203–217.

Black, William and Philip Neal. 1996. Using ALICE to analyse a software manual corpus. In R. Sutcliffe et al (1996), 47–56.

Booth, Barbara. 1985. Revising CLAWS. *ICAME Journal* 9:29–35.

Brill, Eric. 1993. *A Corpus-based approach to language learning.* PhD thesis, Department of Computer and Information Science, University of Pennsylvania.

Briscoe, Edward and John Carroll. 1993. Generalised probabilistic LR parsing of natural language (corpora) with unification-based grammars. *Computational Linguistics* 19:25–60.

EAGLES (1996), WWW site for European Advisory Group on Language Engineering Standards, http://www.ilc.pi.cnr.it/EAGLES96/home.html Specifically: Leech, Geoffrey, Ruthanna Barnett and Peter Kahrel, *EAGLES Final Report and guidelines for the syntactic annotation of corpora*, EAGLES Report EAG-TCWG-SASG/1.5.

Eeg-Olofsson, Mats. 1991. *Word-class tagging: Some computational tools*. PhD thesis. Department of Linguistics and Phonetics, University of Lund, Sweden.

Elliott, John and Eric Atwell. 2000. Is there anybody out there?: the detection of intelligent and generic language-like features. In *Journal of the British Interplanetary Society*, 53:1/2, 13–22.

Garside, Roger. 1996. The robust tagging of unrestricted text: the BNC experience. In J. Thomas and M. Short (eds). *Using corpora for language research: studies in the honour of Geoffrey Leech,* 167–180. London: Longman.

Greene, Barbara and Gerald Rubin. 1981. *Automatic grammatical tagging of English*. Providence, R.I.: Department of Linguistics, Brown University.

Greenbaum, Sidney. 1993. The tagset for the International Corpus of English. In C. Souter and E. Atwell (eds). *Corpus-based Computational Linguistics*, 11–24. Amsterdam: Rodopi.

Grefenstette, Gregory. 1996. Using the SEXTANT low-level parser to analyse a software manual corpus. In R. Sutcliffe et al (1996), 139–158.

Hughes, John and Eric Atwell. 1994. The automated evaluation of inferred word classifications. In A. Cohn (ed). *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 535–539. Chichester, John Wiley.

Hughes, John, Clive Souter and Eric Atwell. 1995. Automatic extraction of tagset mappings from parallel-annotated corpora. In *From texts to tags: issues in multilingual language analysis. Proceedings of SIGDAT workshop in conjunction with the 7th Conference of the European Chapter of the Association for Computational Linguistics.* University College Dublin, Ireland.

Johansson, Stig, Eric Atwell, Roger Garside and Geoffrey Leech. 1986. *The Tagged LOB corpus: users' manual*. Bergen University, Norway: ICAME, The Norwegian Computing Centre for the Humanities. Available from http://www.hit.uib.no/icame/lobman/lob-cont.html

Karlsson, Fred, Atro Voutilainen, Juha Heikkilä, and Arto Anttila. 1995. *Constraint Grammar: a language-independent system for parsing unrestricted text.* Berlin: Mouton de Gruyter.

Kytö, Merja and Atro Voutilainen. 1995. Applying the Constraint Grammar parser of English to the Helsinki Corpus. *ICAME Journal* 19:23–48.

Leech, Geoffrey, Roger Garside and Eric Atwell. 1983. The automatic grammatical tagging of the LOB corpus. *ICAME Journal* 7:13–33.

Lin, Dekang. 1994. PRNCIPAR – an efficient, broad-coverage, principle-based parser. *Proceedings of COLING-94, Kyoto.* 482–488.

Lin, Dekang. 1996. Using PRINCIPAR to analyse a software manual corpus. In R. Sutcliffe et al (1996), 103–118.

man 1986. *parts.* The on-line Unix manual.

Manning, Christopher and Hinrich Schutze. 1999. *Foundations of statistical natural language processing*. Cambridge: MIT Press.

Marcus, Mitch, M. Marcinkiewicz, and Barbara Santorini. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics* 19:313–330.

O'Donoghue, Tim. 1991. Taking a parsed corpus to the cleaners: the EPOW corpus. *ICAME Journal* 15:55–62

Oostdijk, Nelleke. 1996. Using the TOSCA analysis system to analyse a software manual corpus. In R. Sutcliffe et al (1996), 179–206.

Osborne, Miles. 1996. Using the Robust Alvey Natural Language Toolkit to analyse a software manual corpus. In R. Sutcliffe et al (1996), 119–138.

Owen, M. 1987. Evaluating automatic grammatical tagging of text. *ICAME Journal* 11:18–26.

Qiao, Hong Liang and Renje Huang. 1998. Design and implementation of AGTS probabilistic tagger. *ICAME Journal* 22: 23–48.

Santorini, Barbara. 1990. *Part-of-speech tagging guidelines for the Penn Treebank project*. Technical report MS-CIS-90-47. University of Pennsylvania: Department of Computer and Information Science.

Sleator, Daniel and Davy Temperley. 1991. *Parsing English with a Link grammar.* Technical Report CMU-CS-91-196. School of Computer Science, Carnegie Mellon University.

Souter, Clive. 1989a. The COMMUNAL project: extracting a grammar from the Polytechnic of Wales corpus. *ICAME Journal* 13:20–27.

Souter, Clive. 1989b *A short handbook to the Polytechnic of Wales Corpus*. Bergen University, Norway: ICAME, The Norwegian Computing Centre for the Humanities. Available from http://kht.hit.uib.no/icame/manuals/pow.html

Sutcliffe, Richard, Heinz-Detlev Koch and Annette McElligott (eds). 1996. *Industrial parsing of software manuals*. Amsterdam: Rodopi.

Sutcliffe, Richard and Annette McElligott. 1996. Using the Link parser of Sleator and Temperley to analyse a software manual corpus. In R. Sutcliffe et al (1996), 89–102.

Taylor, Lolita and Gerry Knowles. 1988. *Manual of information to accompany the SEC corpus: the machine readable corpus of spoken English*. University of Lancaster: Unit for Computer Research on the English Language. Available from http://kht.hit.uib.no/icame/manuals/sec/INDEX.HTM

Teahan, Bill. 1998. *Modelling English text.* PhD Thesis, Department of Computer Science, University of Waikato, New Zealand.

Ting, Christopher and Peh Li Shiuan. 1996. Using a dependency structure parser without any grammar formalism to analyse a software manual corpus. In R. Sutcliffe et al (1996), 159–178.

Voutilainen, Atro and Timo Järvinen. 1996. Using the English Constraint Grammar Parser to analyse a software manual corpus. In R. Sutcliffe et al (1996), 57–88.