Lexicon and grammar in probabilistic tagging of written English.

Andrew David Beale
Unit for Computer Research on the English Language
University of Lancaster
Bailrigg, Lancaster
England LA1 4YT
enb025@uk.ac.lancs.vax1

Abstract

The paper describes the development of software for automatic grammatical analysis of unrestricted, unedited English text at the Unit for Computer Research on the English Language (UCREL) at the University of Lancaster. The work is currently funded by IBM and carried out in collaboration with colleagues at IBM UK (Winchester) and IBM Yorktown Heights. The paper will focus on the lexicon component of the word tagging system, the UCREL grammar, the databanks of parsed sentences, and the tools that have been written to support development of these components. This work has applications to speech technology, spelling correction, and other areas of natural language processing. Currently, our goal is to provide a language model using transition statistics to disambiguate alternative parses for a speech recognition device.

1. Text Corpora

Historically, the use of text corpora to provide empirical data for testing grammatical theories has been regarded as important to varying degrees by philologists and linguists of differing persuasions. The use of corpus citations in grammars and dictionaries pre-dates electronic data processing (Brown, 1984: 34). While most of the generative grammarians of the 60s and 70s ignored corpus data, the increased power of the new technology nevertheless points the way to new applications of computerized text corpora in dictionary making, style checking and speech recognition. Computer corpora present the computational linguist with the diversity and complexity of real language which is more challenging for testing language models than intuitively derived examples. Ultimately grammars must be judged by their ability to contend with the real facts of language and not just basic constructs extrapolated by grammarians.

2. Word Tagging

The system devised for automatic word tagging or part of speech selection for processing running English text, known as the Constituent-Likelihood Automatic Word-tagging System (CLAWS) (Garside et al., 1987) serves as the basis for the current work. The word tagging system is an automated component of the probabilistic parsing system we are currently working on. In word tagging, each of the running words in the corpus text to be processed is associated with a pre-terminal symbol, denoting word class. In essence, the CLAWS suite can be conceptually divided into two phases: tag assignment and tag selection.

```
constable          NNS1 NNS1: NP1:
constant           JJ NN1
constituent        NN1
constitutional     JJ NN1@
construction       NN1
consultant         NN1
consummate         JJ VV0
contact            NN1 VV0
contained          VVD VVN JJ@
containing         VVG NN1%
contemporary       JJ NN1@
content            NN1 JJ VV0@
contessa           NNS1 NNS1:
contest            NN1 VV0@
contestant         NN1
continue           VV0
continued          VVD VVN JB@
contraband         NN1 JJ
contract           NN1 VV0@
contradictory      JJ
contrary           JJ NN1
contrast           NN1 VV0@
```

Figure 1: Section of the CLAWS Lexicon

JB = attributive adjective; JJ = general adjective; NN1 = singular.common noun; NNS1 = noun of style or title; NP1 = singular proper noun; VV0 = base form of lexical verb; VVD = past tense of lexical verb; VVG = *-ing* form of lexical verb; VVN = past participle of lexical verb; %, @ = probability markers; : = word initial capital marker.

Tag assignment involves, for each input running word or punctuation mark, lexicon look-up, which provides one or more potential word tags for each input word or punctuation mark. The lexicon is a list of about 8,000 records containing fields for

(1) the word form

(2) the set of one or more candidate tags denoting the word's word class(es) with probability markers attached indicating three different levels of probability.

Words not in the CLAWS lexicon are assigned potential tags either by suffixlist look-up, which attempts to match end characters of the input word with a suffix in the suffixlist, or, if the input word does not have a word-ending to match one of these entries, default tags are assigned. The procedures ensure that rare words and neologisms not in the lexicon are still given an analysis.

```
.
.
de      NN1
ade     NN1 VV0 NP1:
made    JJ
ede     VV0 NP1:
ide     NN1 VV0
side    NN1
wide    JJ
oxide   NN1
ode     NN1 VV0
ude     VV0
tude    NN1
ee      NN1
free    JJ
fe      NN1 NP1:
ge      NN1 VV0 NP1:
dge     NN1 VV0
ridge   NN1 NP1:
.
.
```

Figure 2: Section of the Suffixlist

Tag selection disambiguates the alternative tags that are assigned to some of the running words. Disambiguation is achieved by invoking one-step probabilities of tag pair likelihoods extracted from a previously tagged training corpus and upgrading or downgrading likelihoods according to the probability markers against word tags in the lexicon or suffixlist. In the majority of cases, this first order Markov model is sufficient to correctly select the most likely sequence of tags associated with the input running text. (Over 90 per cent of running words are correctly disambiguated in this way.) Exceptions are dealt with by invoking a look up procedure that searches through a limited list of groups of two or more words, or by automatically adjusting the probabilities of sequences of three tags in cases where the intermediate tag is misleading.

The current version of the CLAWS system requires no pre-editing and attributes the correct word tag to over 96 per cent of the input running words, leaving 3 to 4 per cent to be corrected by human post-editors.

3. Error Analysis

Error analysis of CLAWS output has resulted, and continues to result, in diverse improvements to the system, from the simple adjustment of probability weightings against tags in the lexicon to the inclusion of additional procedures, for instance to deal with the distinction between proper names and common nouns.

Parts of the system can also be used to develop new parts, to extend existing parts, or to interface with other systems. For instance, in order to produce a lexicon sufficiently large and detailed enough for parsing, we needed to extend the original list of about 8,000 entries to over 20,000 (the new CLAWS lexicon contains about 26,500 entries). In order to do this, a list of 15,000 words not already in the CLAWS lexicon was tagged using the CLAWS tag assignment program. (Since they were not already in the lexicon, the candidate tags for each new entry were assigned by suffixlist lookup or default tag assignment.) The new list was then post-edited by interactive screen editing and merged with the old lexicon.

Another example of 'self improvement' is in the production of a better set of one-step transition probabilities. The first CLAWS system used a matrix of tag transition probabilities derived from the tagged Brown corpus (Francis and Kučera, 1982). Some cells of this matrix were inaccurate because of incompatibility of the Brown tagset and the CLAWS tagset. To remedy this, a new matrix was created by a statistics-gathering program that processed the post-edited version of a corpus of one million words tagged by the original CLAWS suite of programs.

4. Subcategorization

Apart from extending the vocabulary coverage of the CLAWS lexicon, we are also subcategorizing words belonging to the major word classes in order to reduce the over-generation of alternative parses of sentences of greater than trivial length. The task of subcategorization involves:

(1) a linguist's specification of a schema or typology of lexical subcategories based on distributional and

functional criteria.

(2) a lexicographer's judgement in assigning one or more of the subcategory codes in the linguist's schema to the major lexical word forms (verbs, nouns, adjectives).

The amount of detail demarcated by the subcategorization typology is dependent, in part, on the practical requirements of the system. Existing subcategorization systems, such as the one provided in the Longman Dictionary of Contemporary English (1978) or Sager's (1981) subcategories, need to be taken into account. But these are assessed critically rather than adopted wholesale (see for instance Akkerman et al., 1985 and Boguraev et al., 1987, for a discussion of the strengths and weaknesses of the LDOCE grammar codes).

[1] **intransitive verb** : ache, age, allow, care, conflict, escape, fish, occur, reply, snow, stay, sun-bathe, swoon, talk, vanish.

[2] **transitive verb** : abandon, abhor, allow, build, complete, contain, demand, exchange, get, give, house, keep, mail, master, oppose, pardon, spend, strengthen, warn.

[3] **copular verb** : appear, become, feel, get, grow, remain, seem.

[4] **prepositional verb** : abstract, aim, ask, belong, cater, consist, prey, pry, search, vote.

[5] **phrasal verb** : blow, build, cry, dress, ease, farm, fill, hand, jazz, look, open, pop, share, work.

[6] **verb followed by that-clause** : accept, believe, demand, doubt, feel, guess, know, maintain, reckon, require, think.

[7] **verb followed by to-infinitive** : ask, come, dare, demand, fail, hope, intend, need, prefer, propose, refuse, seem, try, wish.

[8] **verb followed by -ing construction** : abhor, begin, continue, deny, dislike, enjoy, keep, recall, remember, risk, suggest.

[9] **ambitransitive verb** : accept, answer, close, compile, cook, develop, feed, fly, move, obey, practice, quit, sing, stop, teach, try.

[A] **verb habitually followed by an adverbial** : appear, come, go, keep, lie, live, move, put, sit, stand, swim, veer.

[W] **verb followed by a wh-clause** : ask, choose, doubt, imagine, know, matter, mind, wonder.

Figure 3: The initial schema of eleven verb subcategories

We began subcategorization of the CLAWS lexicon by word-tagging the 3,000 most frequent words in the Brown corpus (Kučera and Francis, 1967). An initial system of eleven verb subcategories was proposed, and judgements about which subcategory(ies) each verb belonged to were empirically tested by looking up entries in the microfiche concordance of the tagged Lancaster/Oslo-Bergen corpus (Hofland and Johansson, 1982; Johansson et al., 1986) which shows every occurrence of a tagged word in the corpus together with its context.

About 2,500 verbs have been coded in this way, and we are now working on a more detailed system of about 80 different verb subcategories using the Lexicon Development Environment of Boguraev et al. (1987).

5. Constituent Analysis

The task of implementing a probabilistic parsing algorithm to provide a disambiguated constituent analysis of unrestricted English is more demanding than implementing the word tagging suite, not least because, in order to operate in a manner similar to the word-tagging model, the system requires

(1) specification of an appropriate grammar of rules and symbols and

(2) the construction of a sufficiently large databank of parsed sentences conforming to the (optimal) grammar specified in (1) to provide statistics of the relative likelihoods of constituent tag transitions for constituent tag disambiguation.

In order to meet these prior requirements, researchers have been employed on a full-time basis to assemble a corpus of parsed sentences.

6. Grammar Development and Parsed Subcorpora

The databank of approximately 45,000 words of manually parsed sentences of the Lancaster/Oslo-Bergen corpus (Sampson, 1987: 83ff) was processed to show the distinct types of production rules and their frequency of occurrence in the grammar associated with the Sampson treebank. Experience of the UCREL probabilistic system (Garside and Leech, 1987: 66ff) and suggestions from other researchers prompting new rules resulted in a new context-free grammar of about 6,000 productions creating more steeply nested structures than those of the Sampson grammar. (It was anticipated that steeper nesting would reduce the size of the treebank required to obtain adequate frequency statistics.) The new grammar is defined descriptively in a Parser's Manual (Leech, 1987) and formalised as a set of context-free phrase-structure productions.

Development of the grammar then proceeded in tandem with the construction of a second databank of parsed sentences, fitting, as closely as possible, the rules expressed by the grammar. The new databank comprises extracts from newspaper reports dating from 1979-80 in the Associated Press (AP) corpus. Any difficulties the grammarians had in parsing were resolved, where appropriate, by amending or adding rules to the grammar. This methodology resulted in the grammar

being modified and extended to nearly 10,000 context-free productions by December 1987.

```
V'  =>  V
        Od (I) (V)
        Oh (I) (Vn)
        Ob (I) {(Vg)/(Vn)}
```

Figure 4: Fragment of the Grammar from the Parser's Manual

Ob = operator consisting of, or ending with, a form of *be*; Od = operator consisting of, or ending with, a form of *do*; Oh = operator constisting of, or ending with, a form of the verb *have*; V = main verb with complementation; V' = predicate; Vg = an *-ing* verb phrase; Vn = a past participle phrase; () = optional constituents; {/} = alternative constituents.

7. Constructing the Parsed Databank

For convenience of screen editing and computer processing, the constituent structures are represented in a linear form, as strings of grammatical words with labelled bracketing. The grammarians are given print-outs of post-edited output from the CLAWS suite. They then construct a constituent analysis for each sentence on the print-out, either in detail or in outline, according to the rules described in the Parser's Manual, and key in their structures using an input program that checks for well-formedness. The well-formedness constraints imposed by the program are:

(1) that labels are legal non-terminal symbols

(2) that labelled brackets balance

(3) that the productions obtained by the input analysis are contained in the existing grammar.

One sentence is presented at a time. Any errors found by the program are reported back to the screen, once the grammarian has sent what s/he considers to be the completed parse. Sentences which are not well formed can be re-edited or abandoned. A validity marker is appended to the reference for each sentence indicating whether the sentence has been abandoned with errors contained in it.

```
^ Shortages_NN2 of_IO gasoline_NN1 and_CC
rapidly_RR rising_VVG prices_NN2 for_IF
the_AT fuel_NN1 are_VBR given_VVN as_II
the_AT reasons_NN2 for_IF a_AT1 6.7_MC
percent_NNU reduction_NN1 in_II traffic_NN1
deaths_NN2 on_II New_NP1 York_NP1 state_NN1
's_$ roads_NNL2 last_MD year_NNT1 ._.
```

Figure 5: A word-tagged sentence from the AP corpus

AT = article; AT1 = singular article; CC = coordinating conjunction; IF = *for* as preposition; II = preposition; IO = *of* as preposition; MC = cardinal number; MD = ordinal number; NN2 = plural common noun; NNL2 = plural locative noun; NNT1 = temporal noun; NNU = unit of measurement; RR = general adverb; VBR = *are*; $ = germanic genitive marker.

8. Assessing the Parsed Databank and the Grammar

We have written ancillary programs, to help in the development of the grammar and to check the validity of the parses in the databank. One program searches through the parsed databank for every occurrence of a constituent matching a specified constituent tag. Output is a list of all occurrences of the specified constituent together with frequencies. This facility allows selective searching through the databank, which is a useful tool for revising parts of the grammar.

9. Skeleton Parsing

We are aiming to produce a million word corpus of parsed sentences by December 1988 so that we can implement a variant of the CYK algorithm (Hopcroft and Ullman, 1979: 140) to obtain a set of parses for each sentence. Viterbi labelling (Bahl et al., 1983; Forney, 1973) could be used to select the most probable parse from the output parse set. But problems associated with assembling a fully parsed databank are:

(1) speed of production and

(2) matching the parsed databank to an evolving grammar.

In order to circumvent these problems, a strategy of skeleton parsing has been introduced. In skeleton parsing, grammarians enter minimal labelled bracketing by inserting only those labelled brackets that are uncontroversial and, in some cases, by inserting brackets with no labels. The grammar validation routine is de-coupled from the input program so that changes to the grammar can be made without disrupting the input parsing. The strategy also prevents extensive retrospective editing whenever the grammar is modified. Grammar development and parsed databank construction are not entirely independent however. A subset (10 per cent) of the skeleton parses are extracted for comparison with the current grammar, while another subset (1 per cent) is checked by independent grammarians.

Skeleton parsing will give us a partially parsed databank which should limit the alternative parses compatible with the final grammar. We can either assume each parse is equally likely and use the frequency weighted productions generated by the partially parsed databank to upgrade or downgrade alternative parses or we can use a 'restrained' outside/inside algorithm (Baker, 1979) to find the optimal parse.

```
A010   1 v
[S'[Sd[N'[N'&[N Shortages_NN2 [Po of_IO [N'[N gasoline_NN1 N]N']Po]N]
N'&] and_CC [N'+[Jm rapidly_RR rising_VVG Jm][N prices_NN2 [P for_IF
[N'[Da the_AT Da][N fuel_NN1 N]N']P]N]N'+]N']][V'[Ob are_VBR Ob][Vn
given_VVN [P as_II [N'[Da the_AT Da][N reasons_NN2 N]N']P][P for_IF
[N'[D a_AT1 [M 6.7_MC M]D][N percent_NNU reduction_NN1 [P in_II [N'[N
traffic_NN1 deaths_NN2 [P on_II [N'[D[G[N New_NP1 York_NP1 state_NN1
N] 's_$ G]D][N roads_NNL2 N][Q[Nr'[D[M last_MD M]D] year_NNT1 Nr']Q]
N']P]N]N']P]N]N']P]Vn]V']Sd] ._. S']
```

Figure 6: A Fully Parsed Version of the Sentence in figure 5.

D = general determinative element; Da = determinative element containing an article as the last or only word; G = genitive construction; Jm = adjective phrase; M = numeral phrase; N = nominal; N' = noun phrase; N'& = first conjunct of co-ordinated noun phrase; N'+ = non-initial conjunct following a conjunction; Nr' = temporal noun phrase; P = prepositional phrase; Po = prepositional phrase; Q = qualifier; S' = sentence; Sd = declarative sentence.

```
A062  96 v
"_" [S Now_RT ,_, "_" [Si[N he_PPHS1 N][V said_VVD V]Si] ,_, "_" [S&
[N we_PPIS2 N][V are_VBR negotiating_VVG [P under_II [N duress_NN1 N]
P]V]S&] ,_, and_CC [S+[N they_PPHS2 N][V can_VM play_VV0 [P  with_IW
[N us_PPIO2 N]P][P like_ICS [N a_AT1 cat_NN1 [P with_IW [N a_AT1
mouse_NN1 N]P]N]P]V]S+]S] ._. "_"
```

Figure 7: A Skeleton Parsed Sentence.

word tags: ICS = preposition-conjunction; IW = *with, without* as prepositions; PPHS1 = *he, she*; PPHS2 = *they*; PPIO2 = *us*; PPIS2 = *we*; RT = nominal adverb of time; VM = modal auxiliary verb; *hypertags*: S = included sentence; S& = first coordinated main clause; S+ = non-inital coordinated main clause following a conjunction; Si = interpolated or appended sentence.

10. Featurisation

The development of the CLAWS tagset and UCREL grammar owes much to the work of Quirk et al. (1985) while the tags themselves have evolved from the Brown tagset (Francis and Kučera, 1982). However, the rules and symbols chosen have been translated into a notation compatible with other theories of grammar. For instance, tags from the extended version of the CLAWS lexicon have been translated into a formalism compatible with the Winchester parser (Sharman, 1988). A program has also been written to map all of the ten thousand productions of the current UCREL grammar into the notation used by the Grammar Development Environment (GDE) (Briscoe et al., 1987; Grover et al., 1988; Carroll et al., 1988). This is a preliminary step in the task of recasting the grammar into a feature-based unification formalism which will allow us to radically reduce the size of the rule set while preventing the grammar from overgenerating.

```
V      1
                    .
                    .
                    .
        [ VV0* ]                    50   85
        [ VV0* N' ]                800   86
        [ VV0* J ]                  80   87
        [ VV0* P ]                 400   88
        [ VV0* R ]                  80   89
        [ VV0* Fn ]                100   90
                    .
                    .
                    .
```

Figure 8: A Fragment of the UCREL grammar

```
PSRULE V85 : V1 → V.
PSRULE V86 : V1 → V NP.
PSRULE V87 : V1 → V AP.
PSRULE V88 : V1 → V PP.
PSRULE V89 : V1 → V ADVP.
PSRULE V90 : V1 → V V2[FIN].
```

Figure 9: Translation of the Rules in Figure 8 into GDE representation

11. Summary

In summary, we have a word tagging system that requires minimal post-editing, a steadily accumulating corpus of parsed sentences and a context-free grammar of about ten thousand productions which is currently being recast into a feature-based unification formalism. Additionally, we have programs for extracting statistical and collocational data from both word tagged and parsed text corpora.

12. Acknowledgements

The author is a member of a group of researchers working at the Unit for Computer Research on the English Language at Lancaster University. The present members of UCREL are Geoffrey Leech, Roger Garside (UCREL directors), Andrew Beale, Louise Denmark, Steve Elliott, Jean Forrest, Fanny Leech and Lita Taylor. The work is currently funded by IBM UK (research grant: 823105) and carried out in collaboration with Claire Grover, Richard Sharman, Peter Alderson, Ezra Black and Frederick Jelinek of IBM.

13. References

Erik Akkerman, Pieter Masereeuw and Willem Meijs (1985). 'Designing a Computerised Lexicon for Linguistic Purposes', ASCOT Report No. 1, CIP-Gegevens Koninklijke Bibliotheek, Den Haag, Netherlands.

Lalit R. Bahl, Frederick Jelinek and Robert L. Mercer (1983). 'A Maximum Likelihood Approach to Continuous Speech Recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-5, No. 2, March 1983.

J. K. Baker (1979). 'Trainable Grammars for Speech Recognition,' *Proceedings of the Spring Conference of the Acoustical Society of America.*

Bran Boguraev, Ted Briscoe, John Carroll, David Carter and Claire Grover (1987). 'The Derivation of a Grammatically Indexed Lexicon from the Longman Dictionary of Contemporary English', *Proceedings of ACL-87*, Stanford, California.

Ted Briscoe, Claire Grover, Bran Boguraev, John Carroll (1987). 'A Formalism and Environment for the Development of a Large Grammar of English', *Proceedings of IJCAI*, Milan.

Keith Brown (1984). *Linguistics Today*, Fontana, U.K.

John Carroll, Bran Boguraev, Claire Grover, Ted Briscoe (1988). 'The Grammar Development Environment User Manual', Cambridge Computer Laboratory Technical Report 127, Cambridge, England.

Roger Garside, Geoffrey Leech and Geoffrey Sampson (1987). *The Computational Analysis of English: A Corpus-Based Approach*, Longman, London and New York.

Claire Grover, Ted Briscoe, John Carroll, Bran Boguraev (1988). 'The Alvey Natural Language Tools Project Grammar: A Wide-Coverage Computational Grammar of English', *Lancaster Papers in Linguistics* 47, Department of Linguistics, University of Lancaster: March 1988.

G. Forney, Jr. (1973). 'The Viterbi Algorithm', *Proc. IEEE*, Vol 61: March 1973, pp. 268-278.

W. Nelson Francis and Henry Kučera (1982). *Frequency Analysis of English Usage: Lexicon and Grammar*, Houghton Mifflin, Boston.

Knut Hofland and Stig Johansson (1982). *Word Frequencies in British and American English*, Norwegian Computing Centre for the Humanities, Bergen: Longman, London.

John E. Hopcroft and Jeffrey D. Ullman (1979). *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, Reading, Mass.

Stig Johansson, Eric Atwell, Roger Garside and Geoffrey Leech (1986). 'The Tagged LOB Corpus Users' Manual,' Norwegian Computing Centre for the Humanities, Bergen.

Henry Kučera and W. Nelson Francis (1967). *Computational Analysis of Present-day American English*, Brown University Press, Providence, Rhode Island.

Geoffrey Leech (1987). 'Parsers' Manual', Department of Linguistics, University of Lancaster.

Longman Dictionary of Contemporary English (1978), second edition (1987), Longman Group Limited, Harlow and London, England.

Randolph Quirk, Sidney Greenbaum, Geoffrey Leech and Jan Svartvik (1985). *A Comprehensive Grammar of the English Language*, Longman Inc., New York.

Naomi Sager (1981). *Natural Language Information Processing*, Addison-Wesley, Reading, Mass.

Geoffrey Sampson (1987). 'The grammatical database and parsing scheme' in Garside, Leech and Sampson, pp 82-96.

Richard A. Sharman (1988). 'The Winchester Unification Parsing System', *IBM UKSC Report 999*: April 1988.