Universität Regensburg
Philosophische Fakultät IV
Institut für Anglistik/Amerikanistik

THE TEXTUAL DIMENSION *INVOLVED-INFORMATIONAL*: A CORPUS-BASED STUDY

Magisterarbeit
Sprachwissenschaft

vorgelegt von:
Marc Reymann
Regensburg, Juni 2002

Erstgutachter: Prof. Dr. Roswitha Fischer

Zweitgutachter: Prof. Dr. Rainer Hammwöhner

Table of Contents

1. Introduction

In the study at hand, algorithms and their application to Corpora of the English Language will be presented. Not too long ago, corpus analysis had been a long and tedious procedure which yielded only vague results, as the amount of analyzed data was limited by the manual approach. With the advent and rise of computers in linguistics, the new field of computational linguistic evolved, providing a solid tool for analyzing vast amounts of text. In this context, Oliver Mason rightly remarks:

> "Corpus linguistics is all about analyzing language data in order to draw conclusions about how language works. To make valid claims about the nature of language, one usually has to look at large numbers of words, often more than a million. Such amounts of text are clearly outside the scope of manual analysis, and so we need the help of computers." (Mason, 2000: 3)

A study predestined to be exhaustively transferred to computer-aided linguistics is presented in Douglas Biber's *Variation across speech and writing* (1988) which describes a way to establish a general typology of English texts. In this study, Biber derives a so-called multi-dimensional (MD) approach to typology, which he established as follows:
A review of previous research on register variation provided him with a wealth of linguistic features (cf. Biber 1989: 8f) that may occur with varying frequency in both spoken and written English. Biber explains the selection of used linguistic features as follows:

> "For the purpose of this study, previous research was surveyed to identify potentially important linguistic features – those that have been associated with particular communicative functions and therefore might be used to differing extents in different types of text." (Biber 1988: 72)

As opposed to prior studies, Biber does not concentrate on a fixed set of features and marks them as being typical for a specific genre or text type. He rather uses statistical procedures to extract co-occurring features. Here he adds:

> "No a priori commitment is made concerning the importance of an individual linguistic feature or the validity of a previous functional

> interpretation during the selection of features. Rather, the goal is to include the widest possible range of *potentially* important linguistic features." (ibid.)

Biber collected texts and converted them into a form that can be processed by a computer, counting occurrences of these features in the texts using computer programs written in the programming language PL/1. After normalization and standardization of the resulting figures, he applied a factor analysis and cluster analysis to determine sets of features that co-occur with high frequency in the analyzed texts. The eventual clustering of features leads to the interpretation of clusters as *textual dimensions* that share communicative functions. Dimensions that have features, which occur in complementary distribution are defined as a 'scale' (e.g. Involved vs. Informational). In his approach, Biber defines the following five dimensions (cf. Biber 1989: 10):

1. Involved versus informational production
2. Narrative versus nonnarrative concerns
3. Elaborated versus situation-dependent reference
4. Overt expression of persuasion
5. Abstract versus nonabstract style

This study will concentrate on Biber's dimension 1 "Involved versus informational production" and is divided into two main chapters.

The first chapter deals with the establishing of a completely automatic and modularized computer system written in the programming language PERL, that is able to process any given 'raw' text and produce CSV (comma separated values) files of feature occurrences of the 30 features listed by Biber (1989: 8).
The desired input for the system is 'raw' text, which means that the text should not contain any annotations like most available text corpora of English have.
In a first step, the text is tagged by a freely available decision-tree based Part-of-Speech tagger that is capable of tokenizing the text input thus allowing omission of a separate tokenizer. Moreover, the tagger is also capable of producing base forms (lemmas) of the respective word, which will – as we will see – greatly facilitate the parsing of linguistic features.

To produce more accurate parsing results, it is necessary to give more exact definitions of feature patterns than the relatively vague ones given by Biber (1988: 223-245). These definitions will be retrieved from the CGEL (Quirk 1985) and the Longman Grammar of Spoken and Written English (Biber et al. 1999). In some cases the definitions differ quite a lot, or the patterns given by Biber can simply be not applied on the tagged texts. If such problems appear, they will be discussed in the respective section and in each case the way in which they are overcome will be described.
After the feature is adequately defined, the grammatical pattern as defined in Biber (1988: 223-245) and its PERL translation will be listed, followed by a matching example from the corpora examined in chapter two, whenever appropriate.

Having developed this system, the next part of the study will describe its application on text corpora of English, such as the commonly used LOB/FLOB and BROWN/FROWN corpus pairs as representatives of written English, and the less commonly analyzed corpora of spoken English SEC and COLT. To be used as valid input files for the developed system, these corpora have to be stripped of any additional information (e.g. POS tags) with the help of other PERL programs.

Once the system is applied, the resulting pattern occurrence figures will be briefly analyzed by comparing them with the findings in the LSWE corpus. Since we will be comparing the frequency values for different texts, it is sufficient to apply a simple normalization procedure to the figures. They will not be processed by a standardization algorithm, because this is clearly beyond the scope of this study and would only have to be applied “so that the values of features are comparable.” (Biber 1988: 94). For an evaluation of the accuracy of the system I will present a general overview of the findings, followed by brief observations on the differences between spoken vs. written and AmE vs. BrE corpora. Eventually, a diachronic view on the figures will provide an account for language change.

2. Methods and Algorithms

2.1. Tokenizing and Tagging

Starting from raw untagged and untokenized texts, we first need a program which is able to tokenize these texts because most POS taggers expect text as a list of tokens that are separated by newline. The tokenizer decides what can be considered a 'word' according to orthographic features like e.g. separation of sentence punctuation, the recognition of figures and the canceling of separations at the end of lines.

My first attempt at tokenizing and tagging was done with the tokenizer available at <http://www.phrasys.com/software/dist/003/index.html> and Oliver Mason's QTAG Java tagger from <http://www.english.bham.ac.uk/staff/oliver/software/tagger/>. The results of this pair of programs were rather disappointing mostly because of the low accuracy of the resulting output and the inability of the tagger to assign lemmas to words. The general problem of automatic part-of-speech tagging is illustrated here:

> "Word forms are often ambiguous in their part-of-speech (POS). The English word form *store* for example can be either a noun, a finite verb or an infinitive. In an utterance, this ambiguity is normally resolved by the context of a word: e.g. in the sentence 'The 1977 PCs could store two pages of data.', *store* can only be an infinitive. The predictability of the part-of-speech from the context is used by automatic part-of-speech taggers."[1]

After some more unsatisfactory attempts to find a suitable tokenizer/tagger pair, I finally found TreeTagger by Dr. Helmut Schmid[2] from the Institute of Natural Language Processing at the University of Stuttgart.

According to Schmid, the TreeTagger is based on "a new probabilistic tagging method [...] which avoids problems that Markov Model based taggers face, when they have to estimate transition probabilities from sparse data."[3] In this new approach, "transition probabilities are estimated using a decision tree [...] [and] a

[1] <http://www.ims.uni-stuttgart.de/ftp/pub/corpora/tree-tagger1.pdf>
[2] <http://www.ims.uni-stuttgart.de/~schmid/>
[3] <http://www.ims.uni-stuttgart.de/ftp/pub/corpora/tree-tagger1.pdf>

part-of speech tagger [...] has been implemented which achieves 96.36% accuracy on Penn-Treebank data which is better than that of a trigram tagger (96.06%) on the same data." (ibid.)

Apart from the high tagging accuracy, TreeTagger can handle untokenized input files. Therefore no additional software is needed for text tokenization. Also TreeTagger produces lemmas of every tagged word, which greatly facilitates the creation of search algorithms (cf. 3.1. Private Verbs).

The software license of TreeTagger grants the user "...the rights to use the TreeTagger software for evaluation, research and teaching purposes."[4] However, it is not allowed to use the system for commercial purposes.

TreeTagger (we use version 3.1) is available in precompiled binary form for Sun's Solaris operating system as well as for Linux on i386 architecture. For the system designed in my study we will use RedHat Linux, since both the tagger and the PERL programming language can be used on one single operating system.

Since the tagger is also available with language packs other than English (such as French and Greek), the modularized system developed in this study can be adapted to other languages.

Here is an example of TreeTagger's capabilities:

Input: `Correspondents seeking access to the PNC meetings were required, the other day, to fill in forms to apply for permission from the PLO.` *`(from the SEC corpus)`*

Output:

Correspondents	NNS	correspondent
seeking	VBG	seek
access	NN	access
to	TO	to
the	DT	the
PNC	NP	PNC
meetings	NNS	meeting
were	VBD	be
required	VBN	require
,	,	,
the	DT	the
other	JJ	other
day	NN	day
etc…		

The automatic tagging of all raw text files in a directory is done by a simple shell script that writes the tagged files into a separate directory:

```
#!/bin/sh

echo BE SURE TO ADD tree-tagger cmd and bin directories to your PATH

for i in result/rawtext/*.txt; do
        BASENAME=`basename $i .txt`
        tree-tagger-english "$i" > "result/tagged/$BASENAME.txt"
done
```

2.2. List of Tags

In order to develop PERL modules that are as close to Biber's guidelines (1988: 223-245) as possible, it is essential that we keep in mind the tags that TreeTagger produces as well as Biber's notation of 'constituents' (1988: 222).

1. CC Coordinating conjunction
2. CD Cardinal number
3. DT Determiner
4. EX Existential *there*
5. FW Foreign word
6. IN Preposition or subordinating conjunction
7. JJ Adjective
8. JJR Adjective, comparative
9. JJS Adjective, superlative
10. LS List item marker
11. MD Modal
12. NN Noun, singular or mass
13. NNS Noun, plural
14. NP Proper noun, singular
15. NPS Proper noun, plural
16. PDT Predeterminer
17. POS Possessive ending
18. PP Personal pronoun
19. PP$ Possessive pronoun
20. RB Adverb
21. RBR Adverb, comparative
22. RBS Adverb, superlative
23. RP Particle
24. SYM Symbol
25. TO *to*
26. UH Interjection
27. VB Verb, base form

[4] <http://www.ims.uni-stuttgart.de/~schmid/Tagger-Licence>

28. VBD Verb, past tense
29. VBG Verb, gerund or present participle
30. VBN Verb, past participle
31. VBP Verb, non-3rd person singular present
32. VBZ Verb, 3rd person singular present
33. WDT Wh-determiner
34. WP Wh-pronoun
35. WP$ Possessive wh-pronoun
36. WRB Wh-adverb

2.3. List of Constituents

Taken from Biber (1988: 222)

+ : used to separate constituents
(): marks optional constituents
/: marks disjunctive options
xxx: stands for any word
: marks a word boundary
T#: marks a 'tone unit' boundary
DO: *do, does, did, don't, doesn't, didn't, doing, done*
HAVE: *have, has, had, having, -'ve#, -'d#, haven't, hasn't, hadn't*
BE: *am, is, are, was, were, being, been, -'m#, -'re#, isn't, aren't, wasn't, weren't*
MODAL: *can, may, shall, will, -'ll#, could, might, should, would, must, can't, won't, couldn't, mightn't, shouldn't, wouldn't, mustn't*
AUX: MODAL/DO/HAVE/BE/-'s
SUBJPRO: *I, we, he, she, it, they* (plus contracted forms)
OBJPRO: *me, us, him, them* (plus contracted forms)
POSSPRO: *my, our, your, his, their, its* (plus contracted forms)
REFLEXPRO: *myself, ourselves, himself, themselves, herself, yourself, yourselves, itself*
PRO: SUBJPRO/OBJPRO/POSSPRO/REFLEXPRO/*you/her/it*
PREP: prepositions (e.g. *at, among*)
CONJ: conjuncts (e.g. *furthermore, therefore*)
ADV: adverbs
ADJ: adjectives
N: nouns
VBN: any past tense or irregular past participial verb
VBG: *-ing* form of verb
VB: base form of verb
VBZ: third person, present tense form of verb
PUB: 'public' verbs
PRV: 'private' verbs
SUA: ‚suasive' verbs
V: any verb
WHP: WH pronouns – *who, whom, whose, which*
WHO: other WH words – *what, where, when, how, whether, why, whoever,*

whomever, whichever, wherever, whenever, whatever, however
ART: articles – *a, an, the, (dhi)*
DEM: demonstratives – *this, that, these, those*
QUAN: quantifiers – *each, all, every, many, much, few, several, some, any*
NUM: numerals – *one … twenty, hundred, thousand*
DET: ART/DEM/QUAN/NUM
ORD: ordinal numerals – *first … tenth*
QUANPRO: quantifier pronouns – *everybody, somebody, anybody, everyone, someone, anyone, everything, something, anything*
TITLE: address title
CL-P: clause punctuation (‘.’, ‘!’, ‘?’, ‘:’, ‘;’, ‘-‘)
ALL-P: all punctuation (CL-P plus ‘,’)

2.4. Wordlists

Since not every abbreviation used by Biber has an equivalent tag in TreeTagger’s Penn-Treebank tagset we have to define further PERL modules that import additional wordlists. In the following, the module for address titles will serve as an example for these lists:

```
sub TITLE {
        open (WORDLIST, "<../wordlists/TITLE.txt");
        while (<WORDLIST>) {
                chop ($_);
                $TITLE{$_} = $_;
        }
        close WORDLIST;
}
1;
```

The module `TITLE.pm` opens the text file `TITLE.txt`, which holds a list of personal names as found in Quirk (1985: 291):

```
Mr., Mr, Mrs., Mrs, Ms, Ms., Dr., Dr, Private, Captain, Lord, Lady,
General, Professor, Cardinal, Inspector, Chancellor, Governor, Judge
```

It has to be noted that for easier file handling, the listed items are actually separated by a carriage return and not by a comma. In the same manner, the following modules read in other wordlists that are used in pattern definitions:

```
ADVSUBORD.pm, ALLP.pm, CLP.pm, CONJ.pm, DEM.pm, DEM_AND_THAT.pm,
DISCPART.pm, FIRSTPERS.pm, PLCADV.pm, PREP.pm, PRV.pm, PUB.pm,
SECPERS.pm, SUA.pm, SUBJPRO.pm, TIMADV.pm, TITLE.pm, WHO.pm, WHP.pm
```

2.5. The module `UTILS.pm`

This module employs three subroutines that are used in almost everyone of the following PERL modules that search for patterns:

The subroutine PARSE:

```
sub PARSE {
        @ar = split(/\t/, $lines[$i]);
        $word_1 = $ar[0]; $tag_1 = $ar[1]; $lemma_1 = $ar[2];
        chomp($lemma_1);

        @ar = split(/\t/, $lines[$i+1]);
        $word_2 = $ar[0]; $tag_2 = $ar[1]; $lemma_2 = $ar[2];
        chomp($lemma_2);

        @ar = split(/\t/, $lines[$i+2]);
        $word_3 = $ar[0]; $tag_3 = $ar[1]; $lemma_3 = $ar[2];
        chomp($lemma_3);

        @ar = split(/\t/, $lines[$i+3]);
        $word_4 = $ar[0]; $tag_4 = $ar[1]; $lemma_4 = $ar[2];
        chomp($lemma_4);

        @ar = split(/\t/, $lines[$i+4]);
        $word_5 = $ar[0]; $tag_5 = $ar[1]; $lemma_5 = $ar[2];
       chomp($lemma_5);

        @ar = split(/\t/, $lines[$i-1]);
        $word_minus1 = $ar[0]; $tag_minus1 = $ar[1]; $lemma_minus1 =
$ar[2];
       chomp($lemma_minus1);
}
1;
```

This routine grabs six consecutive sets of *word, tag, and lemma* from the tagged input file according to the following paradigm (example from `sec-_-secmpt08.01.txt`):

	word_	tag_	lemma_
minus1	Here	RB	here
1	‘s	VBZ	be|have
2	the	DT	the
3	weather	NN	weather
4	forecast	NN	forecast
5	for	IN	for

This allows for easier translation of the patterns defined in Biber (1988: 222-245) into PERL. These patterns typically consist of a sequence of both ‘constituents’

and tags like e.g. the definition of *be* as a main verb (Biber: 1988: 229):

BE + DET/POSSPRO/TITLE/PREP/ADJ

This pattern is matched if the lemma of the first word is `be` AND the tag of the following word is either exactly `DT` (determiner), `PP$` (possessive pronoun), `IN` (preposition), begins with `JJ` (ANY adverb), or the word following the form of *be* exists in the wordlist for address titles. This is the way how the expression in the respective PERL module (cf. 3.12) has to be read:

```
($lemma_1 =~ /^be$/) && ($tag_2 =~ /^DT$|^PP\$$|^IN$|^JJ/
                      ||  exists $TITLE{$word_2})
```

If the pattern is actually matched the module calls the subroutine `HIT`, which increases the variable `matchcounter` by 1 and prints out the context in which the pattern has been found:

```
sub HIT {
        $matchcounter++;
        print  "$word_minus1  $word_1  $word_2  $word_3  $word_4  $word_5
$word_6 $word_7 $word_8 $word_9 $word_10\n";
}
1;
```

The final task for the module is to print out the total number of pattern occurrences in the input file and concatenate that number to a CSV (comma separated values) file for further evaluation. This is done by calling the routine `MATCHCOUNTER`:

```
sub MATCHCOUNTER {
        print "$matchcounter occurences\n";
        `echo -n ", $matchcounter" >> ../result/statistic.csv`;
}
1;
```

2.6. ‘Pattern 0’

The module `pattern_0.pm` is used to compute the overall wordcount of the input text, which will be needed for the normalization of pattern occurrences. This is done by counting all occurrences of words that are not sentence punctuation:

```
use UTILS;
use ALLP;

sub pattern_0 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &ALLP;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (!        exists $ALLP{$lemma_1} ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

2.7. Starting the Process

The PERL script `tagged2statistic.pl` is used to eventually invoke the whole process, using all PERL modules which are defined below. For usability reasons it also provides basic syntax error handling:

```
#!/usr/bin/perl

use pattern_0;
use pattern_1;
use pattern_2;
…
…
use pattern_30;

$patterntype=lc $ARGV[0] or die "Usage $0 pattern [text]\n";

%corpus=(
        'sec', "Lancaster/IBM Spoken English Corpus",
        'brown', "Brown Corpus, format 1 (written)",
        'frown', "Freiburg-Brown Corpus of American English (written)",
        'lob', "Lancaster-Bergen-Oslo Corpus (written)",
        'llc', "London-Lund Corpus (spoken)",
        'flob', "Freiburg-LOB Corpus of British English",
        'colt', "Bergen Corpus of London Teenage Language (spoken)"
);

%pattern=(
        'pattern_0', "count words",
        'pattern_1', "private verbs",
        'pattern_2', "THAT deletion",
        …
        …
        'pattern_30', "place adverbials"
);

if (!defined($pattern{$patterntype})) {
        print "Invalid pattern type, choose one of the following\n";
```

```
        foreach $c (keys %pattern) {
                printf ("%-8s : %s\n", $c, $pattern{$c});
        }
        die;
}

print "Pattern: $pattern{$patterntype}\n";
&$patterntype;
```

2.8. Processing the Results

Starting from a CSV file that holds the total numbers of occurrences per pattern, we need to apply *normalization*, which means that the total numbers of occurrences will be divided by the number of words in the text. By multiplying the result with 1000, we get another CSV file which holds a list of pattern occurrences per 1000 words. The normalization is done by the script `divide_and_multiply_1000.pl`:

```
#!/usr/bin/perl

open (CSV, "<result/statistic.csv");
$firstline=1;

while (<CSV>) {
        chop ($_);
        if ($firstline eq 0) {
                @splitted = split(/,/, $_);
                print "@splitted[0]";
                for ($i=1; $i<=31;$i++) {
                        if ($i eq 1) {
                                print ", @splitted[$i]";
                        } else {
                                if (@splitted[1] eq " ") {
                                        print ", ";
                                } else {

                                        printf          ",          %f",
@splitted[$i]/@splitted[1]*1000;
                                }
                        }
                }
                print "\n";
        } else {
                $firstline = 0;
                print "$_\n";
        }
}
close CSV;
```

3. Patterns and Modules

To clarify the procedure that eventually leads to the creation of the PERL modules, the first patterns will be exhaustively analyzed, thus giving a profound insight on how the tags produced by TreeTagger, the constituents defined by Biber, and the grammatical definitions of the patterns interact.

3.1. Private Verbs

3.1.1. Definition

Biber's definition of private verbs (1988: 242) refers to Quirk et al. (1985: 1180-1). Described here we find that private verbs are a relatively small and closed set of verbs, which represent a subdivision of the group of factual verbs. These verbs are defined as follows:

"The 'private' type of factual verb expresses intellectual states such as belief and intellectual acts such as discovery. These states and acts are 'private' in the sense that they are not observable: a person may be observed to *assert that God exists,* but not to *believe that God exists*. Belief is in this sense 'private'." (QUIRK: 1985, 1181)

BIBER (1988: 242) only list a small subset of the list found in QUIRK (1985: 1181) as examples for private verbs and although even Quirk refers to its list only as 'examples', it seems to be complete enough to serve as a sufficient search expression to be used in the corpora.

3.1.2. Examples

Here is a small list of examples for private verbs:

accept, anticipate, ascertain, assume, believe, calculate, check, conclude, conjecture, consider, decide, deduce, deem, demonstrate, determine, etc.

3.1.3. Problems

According to Quirk (1985:1181) some of the verbs categorized as 'private' are also a member of the suasive verbs (e.g. *decide, determine, ensure*). Due to the strictly automated and computerized approach towards register analysis there is no viable way do distinguish between these meanings. The deviation caused by this ambiguity can probably be neglected because of the small number of occurrences and the fact that the following co-occurrence patterns search for public, private and suasive verbs together, without distinguishing among them.

Another problem is the –ise/-ize variation. Due to the capability of the TreeTagger to analyze the lemma of an item, this does, if fact, not apply to our analysis:

```
recognized      VBN  recognize
realise         VB   realize
emphasised      VBD  emphasize
```

(Examples from the tagged SEC corpus)

3.1.4. Search Pattern

Given the list of private verbs in Quirk (1985:1181) in their infinitive form, all possible forms in which the verb can occur need to be taken into account. In the case of e.g. *accept* this comprises the suffix '-s' for the 2nd person, present tense (*accepts*), '-ed' for the past tense/ participle (*accepted*) and '-ing' for the gerund/ participle. These suffixes cannot be easily programmed into the search expression, therefore all irregular forms (e.g. *think, thought, thought*) and deviant spellings are manually added whenever appropriate as described in Quirk (1985:100-103):

- doubling of consonants before *-ing* and *-ed*
- deletion of and addition of *-e* (e.g. in *establish* --> *establishes, guess* --> *guesses*)
- treatment of *-y* (e.g. in *fancy* --> *fancies, imply* --> *implies*)

Although all possible forms could be found by sophisticated search expressions considering the above rules, it is easier to rely on the TreeTagger's capability to analyze the lemma of these verbs.

Another important issue is the lexical ambiguity of some items (e.g. "means" can be both a noun and a verb; "accepted" can be both an adjective and the past participle form of a private verb). In these cases, again, the analysis has to rely on the accuracy of the TreeTagger, which provides a set of tags for verbs:

VB verb, base (believe)
VBD verb, past tense (believed)
VBG verb, -ing (believing) or present participle
VBN verb, past participle (believed)
VBP verb, non-3rd person singular present (believe)
VBZ verb, -s (believes)

Thus, an item that is both tagged as a verb AND also belongs to our manually created list of verbs is counted as an occurrence of a private verb.

```
use PRV;
use UTILS;

sub pattern_1 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &PRV;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                ########################################
                if (exists $PRV{$lemma_1}) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

3.1.5. Examples from the corpora in context:

The name presumably derives from the French royal house which never **learned** and never **forgot**; since Bourbon whiskey, though of Kentucky origin, is at least as much favored by liberals in the North as by

conservatives in the South. (BROWN, cat. G)

Most of them are Democrats and nearly all **consider** themselves, and are viewed as, liberals. (BROWN, cat. G)

3.2. *That*-Deletion

3.2.1. Definition

Quirk (1985: 1049) mentions that "[w]hen the *that*-clause is direct object or complement, the conjunction *that* is frequently omitted except in formal use, leaving a zero *that*-clause[...]. It is similarly omitted when a subject *that*-clause (with anticipatory *it*) is extraposed [...]. Biber (1988:244) limits his definition to the subordinator-*that* deletion and establishes the following three rules:

1. PUB/PRV/SUA + demonstrative pronoun/SUBJPRO[5]
2. PUB/PRV/SUA + PRO/N + AUX/V
3. PUB/PRV/SUA + ADJ/ADV/DET/POSSPRO + (ADJ) + N + AUX/V

As with private verbs which were discussed above, the group of public verbs is also a relatively closed set of verbs which "[...] consists of speech act verbs introducing indirect statements[...]." (Quirk, 1985:1181)

Examples: *acknowledge, add, admit, affirm...*

Suasive verbs are defined as verbs that "[...] can be followed by a *that*-clause either with putative *should* [...] or with the mandative subjunctive. A third possibility, a *that*-clause with an indicative verb, is largely restricted to BrE:" (Quirk, 1985:1182)

People are demanding that she *should leave/leave/leaves* the company (ibid.)

Examples: *agree, allow, arrange, ask, beg...*

[5] NOTE: Any unit boundaries occurring in the patterns are omitted, since the desired input for my analysis is raw text.

As already mentioned above, there are several items in these groups that cannot be assigned to belong to one group exclusively. In the case of the search pattern for the *that* deletion however, this does not influence the search results, because here it is simply searched for occurrences of items belonging to ANY of the groups. As with the search for private verbs alone, the program has to look for all grammatical forms (e.g. *-ed, -ing*)

3.2.2. Search Pattern

3.2.2.1. PUB/PRV/SUA + demonstrative pronoun/SUBJPRO

All three of Biber's rules in this section begin with a verb from the 3 groups as defined above. If the program finds such a verb, in this case it has to try to find a demonstrative or subjective pronoun following that verb, in order to fit the rule. Since our tagging program does not have unambiguous tags for these pronouns and they comprise only very few items, manually created lists will again be provided for our program. These are taken from Quirk (1985:345-346):

DEM: *this, that, these, those*
SUBJPRO: *I, we, you, he, she, it, they*

The TreeTagger separates any occurring contraction following these words and thus makes it easy to identify this kind of *that*-deletion.

3.2.2.2. PUB/PRV/SUA + PRO/N + AUX/V

Biber's definition (Biber 1988:222) of the abbreviation 'PRO' comprises personal pronouns in the subjective and objective case, possessive pronouns and reflexive pronouns. In the case of possessive pronouns I assume both determinative (*my*) and independent function (*mine*) have to be taken into account. Biber's list of pronouns is very short and incomplete and is probably only meant to serve as an example, because there are several types of pronouns (e.g. indefinite pronouns, *of*-pronouns, *wh*-pronouns) which can be inserted into the above pattern and still

yield grammatically correct sentences realizing a *that* deletion. Therefore I add all possible pronouns that may follow the PUB/PRV/SUA constituent.

For our search in the corpora two special cases are of interest:

1. Quirk (ibid.) mentions that '*Them* is sometimes replaced by *'em* […] in familiar use'. These occurrences are taken into account by adding the item ‘em to the list of personal pronouns (in the objective case). The TreeTagger also clearly disambiguates the 'em item as personal pronoun.

2. The contracted form of the pronoun '*us*' (*'s*) cannot be disambiguated by TreeTagger. There is probably no case in which such a contraction can occur in sentences complying to the above rule, and if so, it is presumably only in very small numbers. It is therefore assumed safe to simply omit the item *'s*.

According to the above rule, nouns may appear in the same place as pronouns. In this case it is again relied on the disambiguating skills of the QTAG POS-tagger, which tags nouns as follows:

NN: noun, common singular or mass (*action*)
NNS: noun, common plural (*actions*)
NP: noun, proper singular (*Thailand, Thatcher*)
NPS: noun, proper plural (*Americas, Atwells*)

Following the PRO/N pattern, there has to be any full or auxiliary verb (cf. Biber's list of constituents). It is not necessary to further investigate on the structure following these verbs. As soon as one of the following tags is found, the pattern complies to the above rule:

VB: verb, base form (*believe*)
VBD: verb, past tense (*believed*)
VBG: verb, -ing (*believing*)
VBN: verb, past participle (*believed*)
VBP: verb, non-3rd person sing. pres. (believe)
VBZ: verb, 3rd person sing. pres. (*believes*)
MD: modal auxiliary (*might, will*)

Biber explicitly adds the *'s* contraction of '*has*' and '*is*' to these tags. It is important here to distinguish these contractions form the genitive -s form. TreeTagger shows reasonable accuracy in disambiguating contractions and clearly assigns the label VBZ or POS, respectively.

and	CC	and	Many	JJ	Many
that	WDT	that	deaf	JJ	deaf
's	VBZ	be|have	children	NNS	child
it	PP	it	's	POS	's
.	SENT	.	condition	NN	condition
			goes	VBZ	go
			unrecognised	JJR	<unknown>

(Examples from the tagged SEC corpus)

3.2.2.3. PUB/PRV/SUA + ADJ/ADV/DET/POSSPRO + (ADJ) + N + AUX/V

In this case, the tagger identifies adjectives and adverbs. They consist of the following tags:

JJ: adjective, general (*near*)
JJR: adjective, comparative (*nearer*)
JJS: adjective, superlative (*nearest*)

RB: adverb, general (*chronically, deep*)
RBR: adverb, comparative (*easier, sooner*)
RBS: adverb, superlative (*easiest, soonest*)

Biber defines the constituent DET as the sum of articles, demonstratives, quantifiers and numerals (1988: 244). For TreeTagger this translates into the following tags:

DT: determiner, general (*a, an, the, this, that*). This includes articles!
PDT: determiner, pre- (*all, both, half*)
WDT: determiner, wh- (*what, which, whatever, whichever*)
CD: number, cardinal (*four*)

The POSSPRO constituent is tagged by TreeTagger as:

PP$: pronoun, possessive (*my, his*)

3.2.3. Problems

Quirk (1985: 258) mentions that "[i]n addition to this predeterminer function, *all, both,* and *half* as pronouns can take partitive *of*-phrases, which are optional with nouns and obligatory with pronouns..."

In sentences like e.g. *I think half of the time, he was asleep*, the tagger would not recognize the item '*of*' as a part of the determiner '*half*' and thus would not count it as an occurrence of *that deletion*. Generally, the constituent *DET* is not necessarily realized in a single word. It may occur as a series of words (*phrase*) preceding the head, namely: *Predeterminer, central determiner* and *postdeterminer*.

The realization of constituents in phrases rather than in single words can also be observed in the *ADJ* and *ADV*-constituents (LGSWE, 1999:101-2).

Example: *The brave old little man was saved.*

Since our tagging program treats every word separately and cannot recognize phrases, we would have to allow multiple subsequently occurring realizations of all relevant constituents to appear in our search expression, including the item *'of'* (tagged as preposition by TreeTagger), e.g. ADJ+ADJ+ADJ+DET+DET and so forth. However, Biber's account does not mention this kind of pattern, and to keep our algorithms as close to Biber's as possible, we do not allow multiple instances of constituents as well. The resulting PERL module looks like this:

```
use PUB;
use PRV;
use SUA;
use DEM;
use SUBJPRO;
use UTILS;

sub pattern_2 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &PUB;
        &PRV;
        &SUA;
        &DEM;
```

```
        &SUBJPRO;
        for($i = 0; $i <= $#lines; $i++) {
                &PARSE;
                ########################################
                if ((exists $PUB{$lemma_1} || exists $PRV{$lemma_1} ||
exists $SUA{$lemma_1}) && $tag_1 =~ /^V/) {
                        if    (exists    $DEM{$lemma_2}    ||    exists
$SUBJPRO{$lemma_2}) {
                                &HIT;
                        } elsif ($tag_2 =~ /^N|^PP/) {
                                if ($tag_3 =~ /^VB|^MD/) {
                                        &HIT;
                                }
                        } elsif ($tag_2 =~ /^JJ|^RB|^DT|^WDT|^CD|^PP\$/)
{
                                if ($tag_3 =~ /^JJ/ && $tag_4 =~ /^NN/ &&
$tag_5 =~ /^VB/) {
                                        &HIT;
                                } elsif ($tag_3 =~ /^NN/ && $tag_4 =~
/^VB/) {
                                        &HIT;
                                }
                        }
                        ########################################
                }
        }
        &MATCHCOUNTER;
}
1;
```

In line 4 of this module we use the wordlist `DEM.txt`, which excludes the item *that* because otherwise the expression `if (exists $DEM{$lemma_2}…` would also count sentences in which *that* actually occurs and is not omitted.

3.2.4. Examples from BROWN, cat. G:

I suppose [*][6] the reason is a kind of wishful thinking: don't talk about the final stages of Reconstruction and they will take care of themselves.

In spots such as the elbows and knees the second skin is worn off and I realized [*] the aborigines were much darker than they appeared…

I think [*] it is rather foolhardy to trust to luck...

[6] [*] marks the omission of *that*.

3.3. Contractions

3.3.1. Definition

According to the LGSWE (1999:1128), two major classes of contraction can be distinguished in English: verb contraction (e.g. *she's going*) and *not*-contraction (e.g. *couldn't go*). The LGSWE states, that "[v]erb contractions occurs with the primary verbs *be* and *have* as well as with the modal verbs *will* and *would*" and shows the following table:

	present tense			past tense
	1st person	2nd person +3rd person plural	3rd person singular	
be	*am* ~ ***'m***	*are* ~ ***'re***	*is* ~ ***'s***	
have	*have* ~ ***'ve***	*have* ~ ***'ve***	*has* ~ ***'s***	*had* ~ ***'d***
modal verbs	*will* ~ ***'ll***	*would* ~ ***'d***		

This type of contraction may occur after preceding personal pronouns (e.g. *I, she, it*), some adverbs (e.g. *now, here, there*) and, less frequently, with nouns and universal or partitive pronouns (e.g. *everyone, none*). (cf. Hiller, 1983: 16f)

On *not*-contraction, the LGSWE adds that *"Not*-contraction occurs when *not* is reduced and attached to a preceding primary verb (as main verb or auxiliary verb) or modal verb. The resulting negative auxiliary verb is spelled with a final *n't*, as in: *aren't, isn't haven't, didn't, can't, couldn't,* etc."

Hiller (1983: 16) supposes that *not*-contractions of forms like *must, shall, dare,* etc. are of no statistic relevance. Nevertheless, we must include these occurrences in our analysis, to yield comparable results.

3.3.2. Search Pattern

In our case TreeTagger recognizes the *n't*-token as negative marker but tags it as

an adverb (RB). Fortunately, TreeTagger's ability to also determine the lemma of this token (*not*) greatly facilitates the discovering of all *not* contractions. In the case of verb contractions, all items beginning with an apostrophe and being tagged as either VB* (any verb) or MD (this includes modal verbs) will be counted.

Having this in mind, it is clear that we look at the tags assigned to the contracted forms themselves rather than the preceding forms. Thereby we also take into account the *'s* suffixed on nouns as demanded by Biber (1988: 243). Since items tagged as POS (Possessive ending) are not counted, we finally exclude all possessive forms of contractions and comply with Biber's paradigm:

```
use UTILS;

sub pattern_3 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                ########################################
                if ($word_1 =~ /^'/ && $tag_1 =~ /^VB|^MD/) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

Examples (BROWN, cat. G):

That**'s** the law. But what if somebody decides to break it?

The box is internally wired so the door can never be opened without setting off a screeching klaxon ("It**'s** real obnoxious").

"I**'d** wind up full of .38 bullet holes", he said…

3.4. Present-Tense Verbs

For an in-depth analysis of present-tense verbs, we first need to determine the verb categories applicable for this pattern.

3.4.1. Definition:

"Verbs, as a class of words, can be divided into three major categories, according to their function within the verb phrase; we distinguish the open class of FULL VERBS (or lexical verbs) such as LEAVE […] from the closed classes of PRIMARY VERBS (BE, HAVE, and DO […]) and of MODAL AUXILIARY VERBS (*will, might,* etc. […]). Of these three classes, the full verbs can act only as main verbs […], the modal auxiliaries can act only as auxiliary verbs, and the primary verbs can act either as main verbs or as auxiliary verbs."

(Quirk 1985: 96)

According to this account and Biber's definition (Biber 1988: 224), we can first filter out all auxiliary verbs, which cannot function as main verbs (and are correctly identified as modals by the tagger). This leaves us with the obviously relevant class of full verbs and the class of primary verbs, which needs further investigation.
TreeTagger also recognizes the base form of verbs and tags it as VB, so we don't need to differentiate these cases after the tagging process.
The obvious problem for the tagger is that present tense forms of *to be* and *to have* can be constituents of other, non-present tenses, so we have to filter these forms it the search pattern.

3.4.2. Search Pattern

To count full verbs in the present tense, we sum up all occurrences of the VBP and VBZ tags, if the according lemma is NOT `be`, `have` or `do`.
In these cases, we now subdivide:
If the lemma is `be`, count as occurrence if not followed by the word `n't` or `not` (optional) and any other verb.
If the lemma is `have`, count as occurrence if followed by the word `n't` or `not` (optional) and any other verb except its lemma is `get`.

```
use UTILS;

sub pattern_4 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
```

```
        }
        for($i = 0; $i <= $#lines; $i++) {
                &PARSE;
                ######################################
                if ($tag_1 =~ /^VBP|VBZ/) {
                        if ($lemma_1 =~ /^be$|^have$|^do$|^be\|have$/i) {
                                if ($lemma_1 =~ /^be$/) {
                                        if  ($word_2  =~  /^n't$|^not/  &&
$tag_3 !~ /^V/) {
                                                &HIT;
                                        } elsif ($tag_2 !~ /^V/) {
                                                &HIT;
                                        }
                                }
                                if ($lemma_1 =~ /^have$/) {
                                        if ($word_2 =~ /^n't$|^not/) {
                                                if  ($tag_3  =~  /^V/  &&
($lemma_3 =~ /^get$/)) {
                                                        &HIT;
                                                }
                                        }  elsif  ($tag_2  =~  /^V/  &&
($lemma_2 =~ /^get$/)) {
                                                &HIT;
                                        }
                                }
                        } else {
                                &HIT;
                        }
                        ######################################
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.4.3. Examples (BROWN, cat. G)

NORTHERN liberals **are** the chief supporters of civil rights and of integration.

The name presumably **derives** from the French royal house which never learned and never forgot; since Bourbon whiskey, though of Kentucky origin, **is** at least as much favored by liberals in the North as by conservatives in the South.

3.5. 2nd Person Pronouns

3.5.1. Definition

A list of pronouns can be found in Quirk (1985: 346). Biber (1988: 225) restricts the search of 2nd person pronouns to personal and reflexive pronouns including their contracted forms. Possessive forms (*your, yours*) are omitted, hence the following pronouns are of importance:

you, your, yourself, yourselves

3.5.2. Search Pattern

In this case, simply all occurrences of the above words are counted without double checking by referring to their assigned tags.

```
use SECPERS;
use UTILS;

sub pattern_5 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &SECPERS;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                ########################################
                if (exists $SECPERS{$lemma_1}) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

3.5.3. Examples in context (SEC)

'**You** nearly had a fried journalist on your hands there!'

From Berlin to Vladivostok **you** can gaze at appalling statues of …

'How do **you** like **your** knife, sir?'

3.6. *DO* as a Pro-Verb

3.6.1. Definition

Do as a pro-form can substitute verbs, predicates and entire clauses. “[To] distinguish *do* functioning as an operator (where it is not a substitute, but a dummy operator) and *do* functioning as a main verb [, it is important to know,] that the main verb *do* has nonfinite forms (*doing* and *done*) as well as finite forms (*do, does, did*).” (Quirk 1985: 879). Quirk gives the following list (shortened):

DO AS:	FOLLOWED BY:	SUBSTITUTING FOR:	RESTRICTIONS:
operator *do*	-	-	not a substitute but used with quasi-ellipsis
main verb *do* (intransitive)	-	predication	BrE only
main verb *do* (transitive?)	+ *so*	predication	In AmE, and to some extent in BrE, dynamic meaning only
main verb *do* (transitive)	+ *that*	predication	dynamic meaning only
main verb *do* (transitive)	+ *it*	predication	dynamic agentive meaning only

Examples:

> Sam kicked the ball harder than Dennis *did*. (operator, cf. Quirk p. 905)
> He likes cheese and I *do*, too.

According to Biber's search expression (1988: 226), he does not make a distinction as exact as above. For the sake of comparability we will use Biber's definition rather than establishing our own rules.

3.6.2. Search Pattern

We count all forms of *do*, when it does NOT occur in these constructions:

a) DO + (ADV) + V (DO as auxiliary)
b) ALL-P/WHP + DO (DO as question)[7]

In a) the ADV constituent may be omitted altogether or be realized as a simple adverb phrase. Therefore, there can be multiple occurrences of adverbs.

```
use ALLP;
use UTILS;
use WHP;

sub pattern_6 {
```

[7] Here again, tone units are omitted.

```
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &ALLP;
        &WHP;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        ($lemma_1 =~ /^do$/i) &&
                        !(
                                ($tag_2 =~ /^RB/ && $tag_3 =~ /^V/) ||
                                ($tag_2 =~ /^V/) ||
                                (exists  $ALLP{$word_minus1}  ||  exists
$WHP{$lemma_minus1})
                        )
                ) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

3.6.3. Examples (SEC):

But it is a delusion to believe, as some Namibian politicians clearly **do**, that the issue is near the top of the political agenda of the major Western powers.

Also here in its place on the DIYE stage is Mr Micawber's dichotomy: either we-as-a-nation have enough energy or we **don't**.

The more adventurous of you will now be putting your British Telecom applications in the post, if you've not **done so** already.

3.7. Analytic Negation

3.7.1. Definition

Biber (1988: 245) refers to Tottie (1983a), who “distinguishes between synthetic and analytic negation”. Tottie, in return, refers to Poldauf (1964) to explain his terminology: “Poldauf (1964) uses the terms ANALYTICAL and SYNTHETIC to refer to *not*-negation and *no*-negation, respectively.” (Tottie 1983a: 89)

The LGWSE (1995: 88) states that in opposition to *no*-negation which can be expressed by means like pronouns and determiners, “*[n]ot* is in many ways like an adverb, but it has special characteristics which make it natural to single it out as a

unique member of a class by itself. The main use of *not* (and its reduced form *n't*) is to negate a clause […]". This is achieved by inserting the negator after the operator of the verb phrase, which is "normally the word which directly follows the subject" (Quirk: 79).

The LGSWE (1995: 160) gives the following examples of uses of *not*-negation:

If there is no other auxiliary, *do* is obligatorily inserted as operator:

*You **can** do this but you **can't** do that.*

All uses of *be* behave like auxiliaries and require no *do*-insertion:

*It **wasn't** worth our while.*

Exceptions to this rule are negative imperatives like e.g.:

***Don't be** so hard on yourself.*

It has to be mentioned that there are various subdivisions of negations like the distinction between AUXILIARY and MAIN VERB NEGATION as found in Quirk (1985: 794). However, Biber does not employ this level of subdivision.

3.7.2. Search Pattern

In the case of analytic negation, all occurrences of the lemma *not* are counted. This already includes the contracted form *n't* which also produces the lemma *not* when tagged by TreeTagger.

```
use UTILS;

sub pattern_7 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        ($lemma_1 =~ /^not$/i)
                ) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

3.7.3. Examples (SEC)

Protestants, however, are a tiny minority in Argentina, and the delegation **won't** be including a Roman Catholic.

Of course, what concerns church leaders **isn't** necessarily what worries ordinary churchgoers, even less the general public: …

This understanding, say the scientists, is in fact, unscientific, and the reason is, they say, that natural laws do **not** cause or dictate events: they're merely descriptions of what we expect to happen on the basis of previous experience.

3.8. Demonstrative Pronouns

3.8.1. Definition

"Like the definite article and the personal pronouns, demonstratives have definite meaning, and therefore their reference depends on the context shared by speaker/writer and hearer/reader. Also, in the same way, their use may be considered under the headings of SITUATIONAL reference (reference to the extralinguistic situation), ANAPHORIC reference (coreference to an earlier part of the discourse), and CATAPHORIC reference (coreference to a later part of the discourse). [...][T]he part of the text to which coreference is made [is called] the ANTECEDENT." (Quirk 1985: 372)

Table: Demonstrative pronouns (ibid.)

	SINGULAR	PLURAL
'NEAR' REFERENCE	*this* (student)	*these* (students)
'DISTANT' REFERENCE	*that* (student)	*those* (students)

For further analysis of occurrences of *'that'* there are two interesting features:

a) The item *'that'* can also be a relative pronoun, introducing relative clauses like e.g.:

This is not something *that would disturb me anyway*. (cf. Quirk 1985: 366)

It can only occur in "[r]estrictive relative clauses [which] are closely connected to their antecedent or head prosodically, and denote a limitation on the reference of the antecedent [.]" (ibid.)
In these restrictive clauses, it can occur in both the subjective and the objective case, referring to both personal and nonpersonal antecedents.

b) In the objective case, it can also be omitted completely like in e.g.:

I'd like to see the car *which/that/( )* you bought last week. (Quirk 1985: 366)

3.8.2. Search Pattern

However Biber does not further evaluate this second feature and he establishes the following rules:

that/this/these/those + V/AUX/CL-P/T#/WHP/*and*
(where *that* is not a relative pronoun) (see above)

that's

T# + *that*

In the case of T# + *that*, Biber states that these occurrences were "edited by hand to distinguish among demonstrative pronouns, relative pronouns, complementizers, etc.". In this study, however, we cannot refer to prosodic features like tone unit boundaries (this also applies to the first pattern), because we use raw text corpora.

```
use UTILS;
use DEM_AND_THAT;
use CLP;
use WHP;
sub pattern_8 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &DEM_AND_THAT;
```

```
        &CLP;
        &WHP;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        (exists $DEM_AND_THAT{$lemma_1}) &&
                        (        ($tag_2 =~ /^V|^MD/) ||
                                 ($lemma_2                                  =~
/^do$|^have$|^be$|^be\|have$/i) ||
                                (exists $CLP{$word_2}) ||
                                (exists $WHP{$lemma_2}) ||
                                ($word =~ /^and$/)
                        )
                ) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

3.8.3. Examples (FROWN, cat. A):

"There was no point for **those** who wanted to support the president to switch when he wasn't going to win anyway," said Sen. Slade Gorton, R-Wash.

"**This** is an effort to embarrass President Bush 30 days before the election," Senate Minority Leader Bob Dole of Kansas said before the vote.

"**That**'s a matter of honest disagreement, but I just disagree, and history indicates we have had a lot of good commanders-in-chief with no military service," Clinton
said.

3.9. General Emphatics

3.9.1. Definition

Biber's term "general emphatics" is 'general' indeed in the sense that the patterns he gives (p. 241) comprise various grammatical phenomena like e.g. the emphasizer *for sure* (cf. Quirk 1985: 583), the booster *a lot* (cf. Quirk 1985: 591), the intensifying *such* (cf. LGSWE: 282), the auxiliary *do* in emphatic function (cf. LGSWE: 433) or the periphrastic comparison with *more, most*. Here he also lists rather obscure patterns (e.g. *real* + ADJ) whose origin cannot be retrieved from his account.

For the fact that we will also deal with semantically similar but grammatically different features later on (e.g. amplifiers) it is necessary to further define terms like e.g. *emphasizer* and *booster.*

According to Quirk (985: 583) emphasizers are a subset "of subjuncts concerned with expressing the semantic role of *modality* [...] which have a reinforcing effect on the truth value of the clause or part of the clause to which they apply. In adding to the force (as distinct from the degree) of a constituent, emphasizers do not require that the constituent should be gradable. When, however, the constituent emphasized is indeed gradable, the adverbial takes on the force of an intensifier [...]."

This is illustrated by the following sentences:

a) He *really* may have injured innocent people.
b) He may have *really* injured innocent people.

In sentence a) the constituent *may have...* is not gradable as opposed to sentence b), where the constituent *injured ...* is gradable. This means that the second emphasizer *really* belongs to the subcategory of *intensifiers*. This category is again divided into the subsets of *amplifiers* and *downtoners*.

The above-mentioned 'emphatic' *a lot* actually belongs to another subset of *amplifiers* namely the *boosters*. As opposed to the second subset named *maximizers*, "which can denote the upper extreme of the scale [...] BOOSTERS [...] denote a high degree, a high point on the scale. [...] [E]specially boosters [...] form open classes, and new expressions are frequently created to replace older once whose impact follows the trend of hyperbole in rapidly growing ineffectual." (Quirk 1985: 590)

The pattern DO + V describes the auxiliary *do* in emphatic function, which is defined in the LGSWE:
"As an auxiliary verb, emphatic *do* commonly serves a specialized function of emphasizing the meaning of the whole following predicate, whether in declarative

or imperative clauses." (LGSWE: 433)

As for the adverb *just*, the LGSWE lists as much as four different meanings (LGSWE: 522). Two of these describe semantically opposite functions. The adverb might serve to:

a) increase the intensity of a following element: e.g. *just dreadful, just what I wanted*

or, on the contrary, it might also be:

b) decreasing intensity of a following element: e.g. *just 4.5 points down* (ibid.)

3.9.2. Search Pattern

There is no viable way to automatically differentiate between these contrary meanings. Biber does not explicitly say that these cases were analyzed 'by hand' so we will have to assume that it has simply not been done and include all occurrences of *just* in our analysis as well.

Although Biber's 'definition' of emphatics seems to be established rather arbitrarily, we will comply to the following patterns to yield comparable results:

for sure/a lot/such a/ real + ADJ / *so* + ADJ /DO + V/ *just/really/most/more*

```
use UTILS;

sub pattern_9 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if  (               ($word_1  =~  /^for$/i  &&  $word_2  =~
/^sure$/i) ||
                        ($word_1 =~ /^a$/i && $word_2 =~ /^lot$/i) ||
                        ($word_1 =~ /^such$/i && $word_2 =~ /^a$/i) ||
                        ($word_1 =~ /^real$/i && $tag_2 =~ /^JJ/) ||
                        ($word_1 =~ /^so$/i && $tag_2 =~ /^JJ/) ||
                        ($lemma_1 =~ /^do$/i && $tag_2 =~ /^V/) ||
                        ($word_1 =~ /^just$/i) ||
                        ($word_1 =~ /^really/i) ||
```

```
                        ($word_1 =~ /^most/i) ||
                        ($word_1 =~ /^more/i)

                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.9.3. Examples (FROWN, cat. A)

The bill was immediately sent to the House, which voted 308-114 for the override, 26 **more** than needed.

"That's a matter of honest disagreement, but I just disagree, and history indicates we have had **a lot** of good commanders-in-chief with no military service," Clinton said.

Do you think we are **so stupid** we don't know what you've been doing to our community for the past 10 years?

3.10. 1st Person Pronouns

3.10.1. Definition

As with 2nd person pronouns (cf. 3.5.), a list of pronouns can be found in Quirk (1985: 346).

3.10.2. Search Pattern

Biber (1988: 225) lists the items *I, me, we, us, my, our, myself, ourselves* and explicitly mentions that contracted forms are also counted. Since TreeTagger is very accurate in tokenizing, these forms are automatically included in the following PERL module:

```
use UTILS;
use FIRSTPERS;

sub pattern_10 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &FIRSTPERS;
```

```
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (         (exists $FIRSTPERS{$lemma_1})
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.10.3. Examples (FROWN, cat. A)

"**I** think **I** was right on principle," Bush said in an interview on ABC's 'Good Morning America.'

We won 35 straight .

James expects the Bears to be as solid as they were last season, when "they played **us** better than any team on **our** schedule."

3.11. Pronoun *IT*

3.11.1. Definition

According to Quirk (1985: 347f) can have both referring and non-referring function: "The neuter or nonpersonal pronoun *it* ('REFERRING *it*') is used to refer not only to inanimate objects […], but also to noncount substances […], to singular abstractions […] and even to singular collections of people […]:"

Referring *it*: She made *some soup* and gave *it* to the children. (ibid.)

When *it* has non-referring function it is called 'Prop *it*' (Quirk) or 'Dummy subject' (LGSWE: 125): "Since it is the most neutral and semantically unmarked of the personal pronouns, *it* is used as an 'empty' or 'prop' subject, especially in expressions denoting time, distance, or atmospheric conditions:" (Quirk 1985: 348)

Prop *it*: What time is *it*?; *It*'s warm today. (ibid.)

Biber assumes more frequent use of the pronoun *it* in spoken language and less frequent use in contexts that are more informational due to the fact that *it* "can be

substituted for nouns, phrases and whole clauses" (Biber 1988: 226). This is especially relevant when time is a crucial factor in discourse (which it obviously the case in conversations).

3.11.2. Search Pattern

Since Biber (1988: 226f) gives no detailed description of the search pattern, it is assumed that the PERL module simply has to count all occurrences of the item *it*:

```
use UTILS;

sub pattern_11 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        ($lemma_1 =~ /^it$/i)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.11.3. Examples (FROWN, cat. A)

```
Asked whether it was a blow to Bush for the override to fall so close to
the election, Clinton said, "It won't do him any good."

A Gallup Poll for Life magazine in the spring found that 83 percent of
adults backed the measure and 16 percent opposed it.
```

3.12. *BE* as a Main Verb

3.12.1. Definition

The primary verb *be* can have three different functions: Firstly, it can be an aspect auxiliary (e.g. in: Michael *is* learning Danish.). Secondly, it can be a passive auxiliary (e.g. in: Michael *was* awarded a price.). Thirdly, "[a]s a main verb, *be* is the most important copular verb in English, serving to link the subject noun phrase with a subject predicative or obligatory adverbial […]:" (LGSWE: 428)

SVP_s(NP)	*You drank coffee like it **was** [water].*
SVP_s(AdjP)	*The odds **are** [favorable enough].*
SVA_c	*Well that's how we got acquainted so well because she **was** [in Olie's room] a lot.* (ibid.)

3.12.2. Search Pattern

Having in mind the examples above, it is interesting that Biber does not include nouns occurring after a form of *to be*:

BE + DET / POSSPRO / TITLE / PREP / ADJ (Biber 1988: 229)

However, we translate this pattern into PERL exactly as defined by Biber:

```
use UTILS;
use TITLE;

sub pattern_12 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &TITLE;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (         ($lemma_1 =~ /^be$/) &&
                        ($tag_2 =~ /^DT$|^PP\$$|^IN$|^JJ/ ||
                        exists $TITLE{$word_2})
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.12.3. Example

Among those voting to override in the Senate **was** Democratic vice presidential nominee Al Gore, a co-author of the bill. (FROWN, cat. A)

Mhm. Which **is** a major bummer. ... Why haven't youse got a part in, why are you are, auditioning for West Side Story? (COLT)

No no no no no, okay, if he **was** a woman, Alex wants to marry her, carry on, come on, anyway carry on. (COLT)

3.13. Causative Subordination

3.13.1. Definition

"**Subordinators**, or **subordinating conjunctions**, are words which introduce (mainly finite) dependent clauses. Grammatically, subordinators have a purely syntactic role, and this distinguishes them from other clause initiators (such as *wh*-words), which can also have a role as subject, object, adverbial, etc." (LGSWE: 85)

Biber (1988: 236) states that *"[b]ecause* is the only subordinator to function unambiguously as a causative adverbial. Other forms, such as *as*, *for*, and *since*, can have a range of functions, including causative." The LGSWE (1999: 87) gives an informative overview of these ambiguous functions:

form	Sub.	Prep.	Adv.	examples
for	x			*I concede the point,* ***for*** *I have stated it many times in the past.*
		x		*Oh we're quite happy to rent* ***for*** *a while.*
like	x			*Here today and gone today,* ***like*** *I said.*
		x		***Like*** *many marine painters he had never been at sea.*
since	x			*But this day is something I've dreamed of* ***since*** *I was a kid.*
		x		***Since*** *Christmas, sales have moved ahead.*
			x	*She had not heard one word from him* ***since****.*
though	x			*She had never heard of him,* ***though*** *she did not say so.*
			x	*That's nice* ***though*** *isn't it?*

3.13.2. Search Pattern

Since *because* is the only unambiguous form of causative subordination, it is obviously predestined to be analyzed in automated corpus linguistics. Thus, the task for the PERL module to perform is to count all occurrences of *because* in the corpora:

```
use UTILS;

sub pattern_13 {
        while (<>) {
```

```
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        ($lemma_1 =~ /^because$/i)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.13.3. Examples (SEC)

To our dismay, not **because** we feared the guerrillas, but **because** the chances of being bombed or being caught in a crossfire had suddenly increased dramatically.

The reverse process was adopted in Rumania **because** it was argued that, if firms got less money for their exports, they would try all the harder to increase their earnings.

3.14. Discourse Particles

3.14.1. Definition

“Discourse markers […] are inserts which tend to occur at the beginning of a turn or utterance, and to combine two roles: (a) to signal a transition in the evolving progress of the conversation, and (b) to signal an interactive relationship between speaker, hearer, and message.” (LGSWE: 1086) Considering ambiguity of discourse markers (which are if fact listed in Biber’s search pattern), the LGSWE adds that “[w]ords and phrases which are discourse markers are often ambiguous, sharing the discourse marker function with an adverbial function. (For example, *now* and *well* are both circumstance adverbs […] as well as discourse markers.)” (ibid.)

Speaking of conjunctive comments, Quirk (1985: 633) adds: “Discourse-initiating items can be less easy to account for plausibly, but it seems significant that such items are usually those that have a well-established conjunctive role in mid-discourse use.”

Biber (1988: 241) lists the words *well, now, anyway, anyhow,* and *anyways*, which, according to their semantic role, fall into these categories (cf. Quirk 1985: 634-636):

now:	resultive, transitional/discoursal
anyhow, anyway, anyways	contrastive/concessive

The item *well* is not listed in Quirk's subdivisions, because it seems to be semantically different. This assumption is supported by the LGSWE (1999: 1086):

"***Well*** is a versatile discourse marker, but appears to have the general function of a 'deliberation signal', indicating the speaker's need to give (brief) thought or consideration to the point at issue. It is a very common turn initiator with a variety of functions […]"

3.14.2. Search Pattern

CL-P / #T + *well* / *now* / *anyway* / *anyhow* / *anyways*[8]

In the corresponding PERL module, whenever the parser hits a clause punctuation that is followed by the discourse markers above, it increases the match counter. This translates as follows:

```
use UTILS;
use CLP;

sub pattern_14 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &CLP;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        (exists $CLP{$word_1} &&
                        $lemma_2                                        =~
/^well$|^now$|^anyway$|^anyhow$|^anyways$/i)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
```

[8] Tone units are again omitted.

```
}
1;
```

3.14.3. Examples

```
Well you seen that er? What you doing your maths then? (COLT)

"You young sods are always in a hurry. I'm all right. Now piss off and
watch where you're going." (FLOB, cat. L)

Now if you just calm down, you'll realise the sense of what I'm saying.
(FLOB, cat. L)
```

3.15. Indefinite Pronouns

3.15.1. Definition

"[Indefinite pronouns] lack the element of definiteness which is found in the personal, reflexive, possessive, and demonstrative pronouns, and to some extent also in the *wh*-pronouns. [… They] are, in a logical sense, QUANTITATIVE: they have universal or partitive meaning, and correspond closely to determiners of the same or of similar form […]." (Quirk 1985: 376)

3.15.2. Search Pattern

The above-mentioned lexical ambiguity is, however, not considered by Biber's definition, and the PERL module simply counts all occurrences of the indefinite pronouns he lists[9] (Biber 1988: 226):

```
use UTILS;

sub pattern_15 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if    (                                    ($lemma_1    =~
```

[9] Biber lists the item *nowhere* and omits the indefinite pronoun *no one*, which is clearly an error and is corrected in the module.

```
/^anybody$|^anyone$|^anything$|^everybody$|^everyone$|^everything$|^nobod
y$|^none$|^nothing$|^somebody$|^someone$|^something$/i) ||
                        ($lemma_1 =~ /^no$/i && $lemma_2 =~ /^one$/i)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.15.3. Examples (FLOB, cat. L)

Following **someone** could mean that **someone** else was following you.

Perhaps Ashton did not know him but **anyone** as open as the old boy had been must arouse suspicion.

"**No one** has sent me to kill you. I say again, I don't even know who you are."

3.16. General Hedges

3.16.1. Definition

Biber's notion of hedges comes closest to Quirk's definition of 'downtoners', which "have a generally lowering effect on the force of the verb or predication and many of them apply a scale to gradable verbs." (Quirk 1985: 597). Biber's list of items mostly belong to two subcategories of downtoners:

a) APPROXIMATORS serve to express an approximation to the force of the verb, while indicating that the verb concerned expresses more than is relevant.
b) COMPROMISERS have only a slight lowering effect and tend, as with a), to call in question the appropriateness of the verb concerned.

(cf. Quirk 1985: 597)

3.16.2. Search Pattern

Biber (1988: 240) gives the pattern: "*at about / something like / more or less / almost / maybe / xxx sort of / xxx kind of* (where xxx is NOT: DET / ADJ /

POSSPRO / WHO – excludes *sort* and *kind* as true nouns)", which translates into PERL as such:

```
use UTILS;
use WHO;

sub pattern_16 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &CLP;
        &WHO;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        $lemma_1 =~ /^almost$|^maybe$/i ||
                        ($lemma_1 =~ /^at$/i && $lemma_2 =~ /^about$/i)
||
                        ($lemma_1   =~   /^something$/i   &&   $lemma_2   =~
/^like$/i) ||
                        ($lemma_1 =~ /^more$/i && $lemma_2 =~ /^or$/i &&
$lemma_3 =~ /^less$/i) ||
                        (($tag_1   !~   /^DT$|^JJ|^PP\$$/   &&   !exists
$WHO{$lemma_1}) &&
                        $lemma_2 =~ /^sort$|^kind$/i &&
                        $lemma_3 =~ /^of$/i)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.16.3. Examples (FLOB, cat. L)

If he had discovered a corpse rather than a fatally injured person **maybe** the police would have been summoned.

The younger woman **almost** imperceptibly cringed.

What **kind of** a person shall I revert to when she's gone?

3.17. Amplifiers

3.17.1. Definition

Biber's list of amplifiers mostly belongs to the group of intensifier subjuncts, which "are broadly concerned with the semantic category of DEGREE […]." (Quirk 1985: 589). Semantically, together with *general emphatics* and *hedges* (cf. patterns 9

and 16), they are part of the same group of adverbs. Amplifiers are a subdivision of intensifier subjuncts that are further subdivided into *maximizers* (e.g. *absolutely, altogether, completely*) and *boosters* (e.g. *badly, bitterly, deeply*). “Denoting the upper extreme of the scale” (Quirk 1985: 590), Biber’s ‘amplifiers’ belong to the first group of maximizers except for the item *very*, which can be used to premodify these maximizers.

3.17.2. Search Pattern

Biber lists the items *absolutely, altogether, completely, enormously, entirely, extremely, fully, greatly, highly, intensely, perfectly, strongly, thoroughly, totally, utterly* and the aforementioned *very*, which are searched for using the following PERL module:

```
use UTILS;

sub pattern_17 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if    (                                        ($lemma_1    =~
/^absolutely$|^altogether$|^completely$|^enormously$|^entirely$|^extremel
y$|^fully$|^greatly$|^highly$|^intensely$|^perfectly$|^strongly$|^thoroug
hly$|^totally$|^utterly$|^very$/i)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.17.3. Examples (LOB, cat B)

She has put up the value of her money. Certainly, the rise is **very** small.

The West needs to make it **absolutely** clear that the freedom of West Berlin and free access to it are vital interests not to be retreated from in the present state of Europe.

But Katanga has for so long been represented- not **altogether** falsely- as a secure and industrious little state beset by wild and envious politicians that its less agreeable side has been overlooked.

3.18. Sentence Relatives

3.18.1. Definition

Describing the features of relative clauses, the LGSWE (1999:195) notes that "[s]ome types of relative clauses are not used as postmodifiers of nouns. This is true of nominal relative clauses, where the *wh*-word can be regarded as representing both the antecedent and the relativizer […]. It also applies to so-called sentential relative clauses or sentence relatives, introduced by *which* [...]"
In addition to this, Quirk (1985: 1118) observes a syntactic feature that is very important for automatic parsing of sentence relatives:
"Sentential relative clauses parallel nonrestrictive postmodifying clauses in noun phrases in that they are separated by intonation or punctuation from their antecedent. They are commonly introduced by the relative word *which*."

3.18.2. Search Pattern

Complying to Quirk's definition, Biber (1988: 235) establishes the following pattern:

T# / , + *which*

However, this pattern implies two problems for the automated approach of my system: First of all, tone units (T#) cannot be identified by the used tagger and is consequently omitted in all patterns. This technical drawback also applies to Biber's remark that "[t]hese forms are edited by hand to exclude non-restrictive relative clauses." (ibid.). To allow for complete automatic parsing, the following PERL module only counts all occurrences of *which* whenever it follows a comma:

```
use UTILS;

sub pattern_18 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if  (              ($lemma_1  =~  /^,$/i  &&  $lemma_2  =~
/^which$/i)
                ) {
                        &HIT;
```

```
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.18.3. Examples

Hopes will now grow brighter of further international co-operation, **which** is the only way to solve the payments difficulties that upset the Western world. (LOB, cat. B)

Britain and the U.S., **which** have problems with their balances, will gain some immediate help. (LOB, cat. B)

Put it this way, if I came out here, **which** is eleven foot, I'd get killed, right? (COLT)

3.19. WH Questions

3.19.1. Definition

"*Wh*-questions open with a *wh*-word which indicates an element to be specified by the addressee. The rest is taken to be already known. The element to be specified could be a clause element (subject, object, predicative, adverbial) or part of a phrase. *What + do* is used to ask for specification of the verb phrase[.]" (LGSWE: 204). Acting as interrogative clause markers, *wh*-words "are used as pronouns (*who, whom, what, which*), determiners (*what, which, whose*), or adverbs (*how, when, where, why*).

3.19.2. Search Pattern

Biber explicitly excludes *wh*-pronouns from the pattern so that the constituent WHO consists mostly of items belonging to the group of adverbs. Although there is not necessarily a modal auxiliary following the initial *wh*-word Biber seems to add this rule to exclude other grammatical functions of *wh*-words:

CL-P / T# + WHO + AUX

With the tone units again omitted, the pattern begins with any clause punctuation excepting commas, and is followed by a *wh*-word plus an auxiliary verb. This translate into PERL as follows:

```
use UTILS;
use CLP;
use WHO;

sub pattern_19 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &WHO;
        &CLP;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if  (                exists  $CLP{$lemma_1}  &&  exists
$WHO{$lemma_2} &&
                        ($tag_3     =~     /^MD$/     ||     $lemma_3     =~
/^be$|^have$|^do$|^be\|have$/i) &&
                        $word_3 !~ /'/ &&
                        $word_4 !~ /'/
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.19.3. Examples

Why do Americans never remember that we are asleep when they are awake?! (FLOB, cat. R)

Why is adaptive sex determination, such as that found in the silverside, not widespread in many groups of insects? (FROWN, cat. J)

When do we interpret the transference? **When** do we interpret resistance? **How** deeply do we interpret? (FROWN, cat. J)

3.20. Possibility Modals

3.20.1. Definition

"*[M]odality* may be defined as the manner in which the meaning of a clause is qualified so as to reflect the speaker's judgment of the likelihood of the proposition it expresses being true." (Quirk 1985: 219).

It is typical of possibility modals that they "[…] do not primarily involve human control of events, but do typically involve human judgment of what is or is not likely to happen." (ibid.)

The possibility modals *can, could, may, and might* can range along a scale of *permission (INTRINSIC)* and *possibility and ability (EXTRINSIC)*. Quirk (1985: 221f) lists the following examples:

Even expert drivers *can* make mistakes.	(*possibility*)
Can you remember where they live?	(*ability*)
Can we borrow these books from the library?	(*permission*)

3.20.2. Search Pattern

Biber's pattern does not make a distinction between intrinsic and extrinsic modality. Hence, the according PERL module counts all occurrences of *can, could, may, and might*:

```
use UTILS;

sub pattern_20 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        ($lemma_1 =~ /^can$|^may$|^might$|^could$/i)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.20.3. Examples

Details of this dependence **may** provide important clues to the origin and evolution of the universe. (FROWN, cat. J)

Toby always imagined that if Vanner **could** only arrange enough sleep for one month his face would probably look quite different. (FLOB, cat. R)

More important, however, is that the biblical writers themselves thought that events that followed natural laws **could** still be regarded as miraculous. (SEC)

3.21. Nonphrasal Coordination

3.21.1. Definition

“Coordinators, or coordinating conjunctions, are used to build coordinate structures, both phrases and clauses. Unlike prepositions […] and subordinators […], which both mark the following structure as subordinate, they link elements which have the same syntactic role. The main coordinators are *and, but,* and *or*, with a core meaning of addition, contrast, and alternative, respectively.” (LGSWE: 79)

Coordination subdivides into *phrasal* and *clausal* coordination. If the coordinated elements cannot be identified to be extending a simple noun phrase we speak of *clausal coordination*.

3.21.2. Search Pattern

Biber’s co-occurrence patterns catching clausal coordination are defined rather complex (Biber 1988: 245):

a) T# / , + *and* + *it* / *so* / *then* / *you* / *there* + BE / demonstrative pronoun / SUBJPRO
b) CL-P + *and*
c) *and* + WHP / WHO / adverbial subordinator / discourse particle / conjunct

Therefore, the respective PERL module had to be programmed similarly complex:

```
use DEM_AND_THAT;
use SUBJPRO;
use UTILS;
use CLP;
use WHP;
use WHO;
use CONJ;
use ADVSUBORD;
use DISCPART;

sub pattern_21 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
```

```
        &SUBJPRO;
        &DEM_AND_THAT;
        &CLP;
        &WHP;
        &WHO;
        &CONJ;
        &ADVSUBORD;
        &DISCPART;

        for($i = 0; $i <= $#lines; $i++) {
                &PARSE;
                ########################################
                if (        $lemma_1 =~ /^,$/ && $lemma_2 =~ /^and$/i &&
                        $lemma_3 =~ /^it$|^so$|^then$|^you$|^there$/i &&
                        ($lemma_4      =~      /^be$/i      ||      exists
$DEM_AND_THAT{$lemma_4} ||
                         exists $SUBJPRO{$lemma_4})) {
                                &HIT;

                } elsif (exists $CLP{$lemma_1} && $lemma_2 =~ /^and$/i) {
                        &HIT;
                } elsif ($lemma_1 =~ /^and$/i &&
                        (exists $WHP{$lemma_2} ||
                        exists $WHO{$lemma_2} ||
                        exists $ADVSUBORD{$lemma_2} ||
                        (exists $ADVSUBORD{$lemma_2} ||
                        exists $ADVSUBORD{"$lemma_2 $lemma_3"} ||
                        exists  $ADVSUBORD{"$lemma_2  $lemma_3  $lemma_4"})
||
                        exists $DISCPART{$lemma_2} ||
                        (exists $CONJ{$lemma_2} ||
                        exists $CONJ{"$lemma_2 $lemma_3"} ||
                        exists $CONJ{"$lemma_2 $lemma_3 $lemma_4"} ||
                        exists    $CONJ{"$lemma_2    $lemma_3    $lemma_4
$lemma_5"}))) {
                                &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.21.3. Examples

On arrival at Windsor, Nemo was winched into his new more spacious quarters. After a day's, rest he'll move in with a female killer whale called Winnie, **and** it's hoped they'll be able to mate. (SEC)

Policemen stopped me when they saw me in villages, asked me what I was doing **and** where I was from, but they never asked to see even so much as a passport. (SEC)

None the less, the sales conference is a piss-up. **And** why not? (FLOB, cat. R)

3.22. WH Clauses

3.22.1. Definition

Describing *wh*-clauses as objects, Quirk (1985: 1184) states that "[t]he use of the *wh*-interrogative clause (which generally implies lack of knowledge on the part of the speaker) is particularly common where the superordinate clause is interrogative or negative. On the other hand, there are some verbs which themselves express uncertainty, such as *ask* and *wonder*: these occur with the *wh*-clause without this nonassertive constraint." According to the LGSWE (1999: 694), "[t]he verbs that most commonly control *wh*-clauses in post-predicate position can be grouped into six major semantic domains […]":

a) speech act verbs (e.g. *tell, say, explain*)
b) other communication verbs (e.g. *show, write*)
c) cognition verbs (e.g. *know, think about, remember*)
d) perception verbs (e.g. *see, look at*)
e) verbs of attitude and emotion (e.g. *agree with, condemn, like, hate*)
f) aspectual verbs (e.g. *start, stop, finish*)

3.22.2. Search Pattern

Since most of the aforementioned verbs overlap with the group of public, private, and suasive verbs. Biber uses those verbs to capture *wh*-clauses that occur as object complements and correctly adds: "Other WH clauses could not be identified reliably by automatic analysis and so were not counted." (Biber, 1988: 231). As a consequence, his pattern looks like this:

PUB / PRV / SUA + WHP / WHO + xxx
(where xxx is NOT = AUX)[10]

This pattern translates into PERL as follows:

[10] This excludes *wh*-questions as defined in 3.19.

```
use UTILS;
use WHO;
use WHP;
use PUB;
use PRV;
use SUA;
sub pattern_22 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &WHO;
        &WHP;
        &PUB;
        &PRV;
        &SUA;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        (exists $PUB{$lemma_1} ||
                        exists $PRV{$lemma_1} ||
                        exists $SUA{$lemma_1}) &&
                        (exists $WHP{$lemma_2} ||
                        exists $WHO{$lemma_2}) &&
                        $tag_3 !~ /^MD$/ &&
                        $lemma_3 !~ /^be$|^have$|^do$|^be\|have$/i
                ) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

3.22.3. Examples

[…]we get so used to the usual patterns that we forget **how** amazing they are, but at times, they say, God does unusual things too. (SEC)

She was a real treasure, of yeoman stock and clever in all domestic things, a widow who knew **how** to look after the 'boy,' who was the only other occupant of the house when Mr. Evitt had gone. (LOB, cat. G)

"And we can find them there?" Elric knew so much relief he only then realized **how** desperate he had become. (FLOB, cat. M)

3.23. Final Prepositions

3.23.1. Definition

Quirk (1985: 663) notes that “[n]ormally a preposition must be followed by its complement, but there are some circumstances in which this does not happen.”

They list the following cases in which a so-called DEFERMENT of the preposition has to take place:

a) Passive constructions with a prepositional verb where the subject corresponds to the prepositional complement in the active: e.g. Has *the room* been paid *for*?
b) Infinitive clauses with thematization of the prepositional complement: e.g. He's impossible to work *with*.
c) *–ing* clauses with thematization: e.g. He's worth listening *to*.

The LGSWE (1999: 105) adds that "[a] preposition is said to be stranded if it is not followed by its complement or, where the preposition is bound to a preceding verb, by the prepositional object[…]".

3.23.2. Search Pattern

Biber (1988: 244) counts an occurrence of stranded preposition if the following pattern is matched: PREP + ALL-P / T#
Complying with this pattern, the PERL module for stranded prepositions searches for items tagged as preposition by TreeTagger and followed by any kind of sentence punctuation:

```
use UTILS;
use ALLP;

sub pattern_23 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &ALLP;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        $tag_1 =~ /^IN$/ &&
                        exists $ALLP{$lemma_2}
                ) {
                        &HIT;
                }
                ########################################
        }
        &MATCHCOUNTER;
}
1;
```

3.23.3. Examples

He eased down in the wing-armed kitchen chair, one boot still **on**, sinking his chin into the yards of lace swathing his throat. (FLOB, cat. M)

Nor is this to be wondered **at**, for even today, in the 1960s, no cure has been found for the {6tic[sic!] douloureux. (LOB, cat. G)

He could only say both got jeans **on**, you know. (COLT)

3.24. Adverbs

3.24.1. Definition

Functioning as the (optionally modified) head of an adverb phrase, adverbs can be divided into three classes:

a) simple adverbs, *e.g. just, only, well*
b) compound adverbs, *e.g. somehow, somewhere, therefore*
c) derivational adverbs. The majority of derivational adverbs have the suffix –*ly*, by means of which new adverbs are created from adjectives (and participial adjectives). (cf. Quirk 1985: 438)

Other derivational suffixes are e.g. *-wise, -ward, -ways, fashion,* and –*style*.

3.24.2. Search Pattern

Biber (1988: 238) employs a rather awkward algorithm to identify adverbs by counting "[a]ny adverb form occurring in the dictionary, or any form that is longer than five letters and ends in –*ly*." Fortunately, we can rely on TreeTagger's high accuracy to identify adverbs and by doing so also catch forms of adverbs which Biber's algorithm would not be able to find. To yield statistically independent figures Biber excludes all instances of hedges, amplifiers, downtoners, place adverbials, and time adverbials. This is equally done by the respective PERL module:

```
use UTILS;
use WHO;
use PLCADV;
use TIMADV;

sub pattern_24 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &WHO;
        &PLCADV;
        &TIMADV;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        $tag_1 =~ /^RB/ &&

                        !(

                        $lemma_1 =~ /^almost$|^maybe$/i ||
                        ($lemma_1 =~ /^at$/i && $lemma_2 =~ /^about$/i)
||
                        ($lemma_1   =~   /^something$/i   &&   $lemma_2   =~
/^like$/i) ||
                        ($lemma_1 =~ /^more$/i && $lemma_2 =~ /^or$/i &&
$lemma_3 =~ /^less$/i) ||
                        (($tag_1   !~   /^DT$|^JJ|^PP\$$/   &&   !exists
$WHO{$lemma_1}) &&
                        $lemma_2 =~ /^sort$|^kind$/i &&
                        $lemma_3 =~ /^of$/i)

                        ) &&

                        !($lemma_1                                          =~
/^absolutely$|^altogether$|^completely$|^enormously$|^entirely$|^extremel
y$|^fully$|^greatly$|^highly$|^intensely$|^perfectly$|^strongly$|^thoroug
hly$|^totally$|^utterly$|^very$/i)

                        &&

                        !(exists     $PLCADV{$lemma_1}     ||     exists
$TIMADV{$lemma_1})


                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.24.3. Examples

More news about the Reverend Sun Myung Moon, founder of the Unification church, who's **currently** in jail for tax evasion… (SEC)

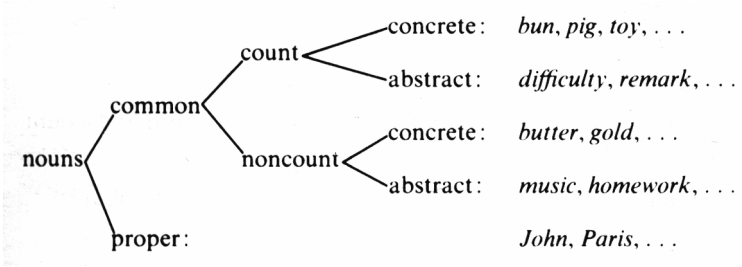
Soon after his return from Europe, Clarence Paget had become **seriously** ill with a supposed abscess on the lungs. (LOB, cat. G)

He's **seriously** deficient in what he should be learning cos, he should know that at least. (COLT)

3.25. Nouns

3.25.1. Definition

As part of a noun phrase the noun can function as subject, object, and complement of clauses and prepositional phrases. Nouns fall into different subclasses. Quirk (1985: 245) gives the following diagram:



3.25.2. Search Pattern

Biber (1988: 228) counts all nouns in the dictionary, excluding nominalizations and gerunds. The PERL module relies on TreeTagger's accuracy of identifying nouns and employs methods to exclude nominalizations (as described in Biber 1988: 227):

```
use UTILS;

sub pattern_25 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (        ($tag_1 =~ /^N/ &&
                        $lemma_1 !~ /tion$|ment$|ness$|ity$/)
                ) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.25.3. Examples

Soon find out, cos you do it all the **morning** and **stuff**. (COLT)

In announcing the **award** in **New York**, the **rector** of the university[11], **Dr Nicholas Argentato**, described **Mr Moon** as a **prophet** of our **time**. (SEC)

3.26. Word Length

3.26.1. Definition

As defined by Biber (1988: 239) 'word length' is the "mean length of the words in a text, in orthographic letters".

3.26.2. Search Pattern

To achieve this, the PERL module adds up the length of all words in the text excluding sentence punctuation. The mean is then calculated by dividing the result by the number of all words in the text excluding punctuation (the total number of words is counted by `pattern_0.pm`; cf. 2.6.).

```
use UTILS;
use ALLP;

sub pattern_26 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &ALLP;
        $matchcounter = 0;
        for($i = 0; $i <= $#lines ; $i++) {
                &PARSE;
                if (!        exists $ALLP{$lemma_1} ) {
                        $matchcounter += length($word_1);
                }
        }
        &MATCHCOUNTER;
}
1;
```

[11] Regarded as nominalization by Biber's algorithm and therefore not counted.

3.27. Prepositions

3.27.1. Definition

“Prepositions are links which introduce prepositional phrases. As the most typical complement in a prepositional phrase is a noun phrase, they can be regarded as a device which connects noun phrases with other structures. Many prepositions in English correspond to case inflections in other languages[12] […]” (LGSWE: 74)

The LGSWE also lists the most common prepositions in English, which are chiefly short, invariable forms: “*about, after, around, as, at, by, down, for, from, in, into, like, of, off, on, round, since, than, to, towards, with, without,* etc.” (ibid.)

3.27.2. Search Pattern

For the module that searches for occurrences of prepositions, Biber’s list (1988: 236f) is used, as it excludes prepositions which function as place or time adverbial, conjunct, or subordinator. These functions are separately defined and would blur any statistic evaluation. Finally, the PERL module looks like this:

```
use UTILS;
use PREP;

sub pattern_27 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &PREP;

        for($i = 0; $i <= $#lines; $i++) {
                &PARSE;
                ########################################
                if (        exists $PREP{$lemma_1}) {
                                &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

[12] In this respect, Finnish is a striking example as most locative references are realized as inflections.

3.27.3. Examples

DAN MORGAN TOLD HIMSELF HE WOULD FORGET Ann Turner. He was well rid **of** her. (BROWN, cat. N)

What a world **of** graceful accomplishment lies **in** a piece **of** finely worked hand-made lace! (LOB, cat. E)

THE Prime Minister's rating is up **in** the polls. (FLOB, cat. B)

3.28. Type/Token Ratio

3.28.1. Definition

As defined by Biber (1988: 238), this ratio is “the number of different lexical items in a text, as a percentage”.

While this is an obviously correct definition, Biber infers that the longer the texts are, the fewer new lexical items can be found in them and thus establishes a rather debatable algorithm. He counts different lexical items found in the first 400 words of a text and then divides that figure by 4 yielding a type/token percentage. While Biber’s reasoning may hold true when analyzing ‘coherent’ texts (e.g. one single academic paper, one complete dialogue), it would heavily bias the analysis of larger text corpora (e.g. BROWN, FLOB, etc.) which consist of broad selections of different text types (cf. the various categories in the BROWN/FROWN and LOB/FLOB corpus pairs).

3.28.2. Search Pattern

Since we mostly analyze larger text corpora, we shall keep the first definition to compute the type/token ratio to remain on a statistically and scientifically sound basis. If it comes to analyzing shorter, coherent texts, the module can, however, be easily modified to comply with Biber’s definition.

```
use UTILS;
use ALLP;

sub pattern_28 {
```

```
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &ALLP;

        for($i = 0; $i <= $#lines; $i++) {
                &PARSE;
                ########################################
                if    (!    exists    $ALLP{$lemma_1}    &&    !    exists
$LIST{lc($word_1)}) {
                        print lc($word_1)."\n";
                        $LIST{lc($word_1)} = lc($word_1);
                }
        }
        $matchcounter = keys( %LIST );
        &MATCHCOUNTER;
}
1;
```

3.29. Attributive Adjectives

3.29.1. Definition

The LGSWE (1999: 515) states that as opposed to predicative adjectives which can function either as subject predicative complementing a copular verb (e.g. *That's **right**. The fans became **restless**…*) or object predicative following a direct object (as in: *I said you've got all your priorities **wrong**.*), "[a]ttributive adjectives modify nominal expressions, preceding the head noun or pronoun. In most cases, they modify common nouns […]" (LGSWE: 510)

Examples for attributive adjectives are e.g.

> Yes, it's a *bad* attitude.
> It's *rusty, knotty* pine with a *huge* fireplace.

3.29.2. Search Pattern

Biber (1988: 238) counts all occurrences of adjectives which are either followed by another adjective or a noun (ADJ + ADJ / N). Hence, the respective PERL module is programmed like this:

```
use UTILS;

sub pattern_29 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        for($i = 0; $i <= $#lines; $i++) {
                &PARSE;
                ########################################
                if ($tag_1 =~ /^JJ/ && $tag_2 =~ /^JJ|^N/) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.29.3. Examples

The **very** word lace has a **charming** derivation, stemming through the **Old French** las, coming from the Latin[13] laqueus, a snare, allied to lacere, to entice. (LOB, cat. E)

The **official** justification is couched in terms of realpolitik. (FLOB, cat. B)

The **easiest** thing would be to sell out to Al Budd and leave the country, but there was a **stubborn** streak in him that wouldn't allow it. (BROWN, cat. N)

3.30. Place Adverbials

3.30.1. Definition

As described in Quirk (1985: 514), adjuncts of space can be subdivided according to their semantic roles into adjuncts concerned with POSITION and DIRECTION. They can have both functions, "[w]here a verb (*e.g. be, live, put*) takes an obligatory predication adjunct […]" (ibid.):

e.g. They are *on the Continent*. She lives *in a cottage*.

Different distribution can be observed "[w]hen a spatial predication adjunct is optional […] [where] it is less likely to express position than DIRECTION:" (ibid.)

[13] Tagged as NP by TreeTagger.

e.g. The children were running very fast *towards the park. / to the swings. / from the school.*

There is a variety of place adverbs listed in Quirk (1985: 516). Many of these adverbs are in fact atrophied prepositional phrases like e.g. *overboard, offshore, underground* etc.

3.30.2. Search Pattern

Using the wordlist created for place adverbials, we count their occurrences in the input text:

```
use UTILS;
use PLCADV;

sub pattern_30 {
        while (<>) {
                $line=$_;
                push(@lines, $line);
        }
        &PLCADV;
        for($i = 0; $i <= $#lines; $i++) {
                &PARSE;
                ########################################
                if (exists $PLCADV{$lemma_1}) {
                        &HIT;
                }
        }
        &MATCHCOUNTER;
}
1;
```

3.30.3. Examples:

His plans and dreams had revolved **around** her so much and for so long that now he felt as if he had nothing. (BROWN, cat. N)

He cleaned his shovel, left it against the fence, picked up his Winchester, and started **downstream**. (BROWN, cat. N)

With a keen, wide chisel the wood is now eased **away above** the toe as at (B). (LOB, cat. E)

4. Applying the System

4.1. Selection on Corpora

For my analysis, I want to do cross-comparisons of both spoken and written English as well as deal with features of corpora from a synchronic and diachronic point of view. Also I want to compare features of different text categories as found in both the BROWN/FROWN and LOB/FLOB corpora. These corpora will also allow for a comparison of registers in BrE and AmE.

Due to the large amount of text, the basis of analysis has to be a strictly computerized and automated system that will not require any further intervention by a human proofreader. This approach, of course, involves an appropriate selection of the text corpora to be analyzed. One problem of most available corpora is the additional information included with the 'raw' text. There are various methods of annotations across the different kinds of corpora (e.g. stress in spoken corpora and part-of-speech tags in manually tagged written corpora), which inhibit accurate automatic tagging by the tagger used in this analysis. In this context the linguist Geoffrey Sampson correctly notes:

> "If one's primary interest is in computer processing of the grammar of spoken English, then the suprasegmental markings get in the way. It is much easier to design software to look up in an electronic dictionary the words of a string such as
>
> ```
> may I ask
> ```
>
> than those of a string
>
> ```
> "^m/\ay* I _ask#
> ```
>
> For grammatical analysis, recording chunks of overlapping turns by different speakers in temporal sequence obscures the grammatical structures used rather than clarifying them."[14]

In this respect, the London-Lund corpus proofed itself to be unusable, since there is no viable way to reconstruct (respectively insert) accurate sentence punctuation.

[14] http://www.cogs.susx.ac.uk/users/geoffs/RChristine.html

In order to permit the above-mentioned comparisons and analyses, I chose the following different English text corpora, which are taken from the second edition of the ICAME CD-ROM (ISBN 82-7283-091-4):

BROWN1	Brown Corpus, format 1 (approx. 1 million words of written AmE, compiled in 1961)
FROWN	Freiburg-Brown Corpus of American English (compiled in 1992 using the same methods)
LOB	Lancaster-Bergen-Oslo Corpus (approx. 1 million words of written BrE, compiled in 1961)
FLOB	Freiburg-LOB Corpus of British English (compiled in 1991 using the same methods)
COLT	Bergen Corpus of London Teenage Language (spoken, orthographic version)
SEC	Lancaster/IBM Spoken English Corpus (approx. 52,000 words of contemporary spoken BrE, started in 1984)

4.2. Preparation of Corpora

The second part of this paper will apply the system developed in the first part on various written and spoken corpora of English. However, most of these corpora contain additional information (e.g. part-of-speech tags, prosodic features, genres, position markers etc.), which is not intended to be a valid input for this specialized system. Therefore, this information has to be stripped off the corpora in order to yield a valid raw text input file. This is done individually for each corpus type, because there is no consistent tagging scheme for all selected corpora. The stripping procedure is invoked through the main PERL Script `corp2txt.pl`.

```
#!/usr/bin/perl
use sec;
use brown;
use frown;
use lob;
use llc;
use flob;
use colt;
```

```
$corptype=lc $ARGV[0] or die "Usage $0 corpustype [corpus]\n";

%corpus=(
        'sec', "Lancaster/IBM Spoken English Corpus",
        'brown', "Brown Corpus, format 1 (written)",
        'frown', "Freiburg-Brown Corpus of American English (written)",
        'lob', "Lancaster-Bergen-Olslo Corpus (written)",
        'flob', "Freiburg-LOB Corpus of British English",
        'colt', "Bergen Corpus of London Teenage Language (spoken)"
);

if (!defined($corpus{$corptype})) {
        print "Invalid corpus type, choose one of the following\n";
        foreach $c (keys %corpus) {
                printf ("%-8s : %s\n", $c, $corpus{$c});
        }
        die;
}

&$corptype;
```

`corp2txt.pl`

The script uses external PERL modules for each corpus type (BROWN, SEC, etc.), which include the regular expressions that perform the actual filtering. These modules will be briefly described here, providing text samples from the respective corpora:

4.2.1. The SEC Corpus

The SEC Corpus (sample input):

```
[001 SPOKEN ENGLISH CORPUS TEXT A01]
[In Perspective]
[Rosemary Hartill]
[Broadcast notes: Radio 4, 07.45 a.m., 24th November, 1984]

Good morning.  More news about the Reverend Sun Myung Moon,
founder of the Unification church, who's currently in jail for
tax evasion: he was awarded an honorary degree last week by
the Roman Catholic University of la Plata in Buenos Aires,
```

Sample from `secapt01.01`

It is obvious, that the SEC files are introduced by 4 lines of specific information about the following text plus an empty line. The text itself is formatted as desired for the next stage except for annotations enclosed in square bracket. Therefore,

the SEC module only has to delete the first 5 lines of all 53 SEC files and delete additional annotations in the texts:

The SEC module `sec.pm`:

```
sub sec {
        while (<>) {
                $line="";
                chop($_);
                $line="$_" unless (1..5);
                push(@lines, $line);
        }
        $text = join "\n", @lines;
        $text =~ s/\[.*?\]//sg;
        print $text;
}
1;
```

While there is input, it prints all lines except for the first 5 lines to the Standard Output (e.g. Screen), using the Range Operator (1..5). After that, it deletes all annotations in square brackets, thus yielding the following output:

```
Good morning.  More news about the Reverend Sun Myung Moon,
founder of the Unification church, who's currently in jail for
tax evasion: he was awarded an honorary degree last week by
```

4.2.2. The BROWN Corpus

The BROWN corpus (sample input):

```
A01 0010    The Fulton County Grand Jury said Friday an investigation
A01 0020 of Atlanta's recent primary election produced "no evidence" that
A01 0030 any irregularities took place.   The jury further said in term-
end
A01 0040 presentments that the City Executive Committee, which had over-
all
A01 0050 charge of the election, "deserves the praise and thanks of the
```

Sample from `brown1_a.txt`

The example shows, that the BROWN files are separated into 3 columns, the first 2 of which contain additional information about the category and line numbers.

Thus, the task of the BROWN module is to delete the first two columns, leaving raw text for further analysis.

```
sub brown {
        while (<>) {
                $line="";
                $line="$1" if /^[A-Z]\d{2}\s+\d*\s+(.+)$/o;
                push(@lines, $line);
        }
        $text = join "\n", @lines;
        $text =~ s/\[.*?\]//sg;
        $text =~ s/\#.*?\#//sg;
        $text =~ s/_.*?_-//sg;
        $text =~ s/{|}//sg;
        $text =~ s/\*\*h/./sg;
        $text =~ s/&/./sg;
        print $text;
}
1;
```

The BROWN module `brown.pm`

The regular expression in line 4 matches a capital letter directly followed by two digits, a space, 4 digits and another space. The module simply sends parts of the read line to standard output, which do not contain the text matching the regular expression, leaving raw unformatted text. Additional information in the corpus is deleted by the regular expressions in lines 8 to 13.

The module produces the following output:

```
The Fulton County Grand Jury said Friday an investigation
of Atlanta's recent primary election produced "no evidence" that
any irregularities took place.   The jury further said in term-end
presentments that the City Executive Committee, which had over-all
charge of the election, "deserves the praise and thanks of the
```

4.2.3. The FROWN Corpus

The FROWN corpus (sample input):

```
A01   1 <#FROWN:A01\><h_><p_>After 35 straight veto victories, intense
A01   2 lobbying fails president with election in offing<p/>
A01   3 <p_>By Elaine S. Povich<p/>
A01   4 <p_>CHICAGO TRIBUNE<p/><h/>
A01   5 <p_>WASHINGTON - Despite intense White House lobbying, Congress
has
A01   6 voted to override the veto of a cable television regulation bill,
```

```
A01   7 dealing President Bush the first veto defeat of his presidency
just
A01   8 four weeks before the election.<p/>
```

Sample from `frown_a.txt`

The annotation scheme of the FROWN corpus is very similar to that of the BROWN corpus, but there are additional SGML tags (enclosing headers etc.) inserted into the text itself. Since these tags would bias the tokenizing and tagging process, the according PERL module had to be significantly altered:

```
sub frown {
        while (<>) {
                $line="";
                $line="$1" if /^[A-Z]\d{2}\s+\d*\s+(.+)$/o;
                push(@lines, $line);
        }
        $text = join "\n", @lines;
        $text =~ s/<h_>.*?<h\/>//sg;
        $text =~ s/<.*?>//sg;
        $text =~ s/\[|\]//sg;
        print $text;
}
1;
```

The FROWN module `frown.pm`

In this case, line 4 again removes information about category and line numbers while the regular expressions in lines 8 to 10 remove the aforementioned tags and square brackets. This procedure results in the following output:

```
WASHINGTON - Despite intense White House lobbying, Congress has
voted to override the veto of a cable television regulation bill,
dealing President Bush the first veto defeat of his presidency just
four weeks before the election.
```

4.2.4. The LOB Corpus

The LOB corpus (sample input):

```
A01   1 **[001 TEXT A01**]
A01   2 *<*'*7STOP ELECTING LIFE PEERS**'*>
A01   3 *<*4By TREVOR WILLIAMS*>
A01   4    |^A *0MOVE to stop \0Mr. Gaitskell from nominating any more
Labour
A01   5 life Peers is to be made at a meeting of Labour {0M P}s tomorrow.
```

```
A01   6    |^\0Mr. Michael Foot has put down a resolution on the subject
and
A01   7 he is to be backed by \0Mr. Will Griffiths, {0M P} for Manchester
A01   8 Exchange.
```

Example from `lob_a.txt`

As seen in the BROWN and FROWN corpora, the LOB corpus introduces every line with category information and line numbers. Additional information is marked by e.g. backslashes, asterisks and brackets, which has to be deleted by the module[15]:

```
sub lob {
        while (<>) {
                $line="";
                $line="$1" if /^[A-Z]\d{2}\s+\d*\s+(.+)$/o;
                push(@lines, $line);
        }
        $text = join "\n", @lines;
        $text =~ s/\\0//sg;
        $text =~ s/\*\d//g;
        $text =~ s/\{0//sg;
        $text =~ s/\|\^//sg;
        $text =~ s/\^//sg;
        $text =~ s/}//sg;
        $text =~ s/<.*?>//sg;
        $text =~ s/\[.*?\]//sg;
        $text =~ s/\*//sg;
        $text =~ s/\\//sg;
        print $text;
}
1;
```

The LOB module `lob.pm`

After being processed by this module, the LOB corpus looks like this:

```
A MOVE to stop Mr. Gaitskell from nominating any more Labour
life Peers is to be made at a meeting of Labour M Ps tomorrow.
Mr. Michael Foot has put down a resolution on the subject and
he is to be backed by Mr. Will Griffiths, M P for Manchester
Exchange.
```

[15] For a detailed description of the annotation scheme see http://khnt.hit.uib.no/icame/manuals/lob/INDEX.HTM#lob6

4.2.5. The FLOB Corpus

The FLOB corpus (sample input):

```
D01   1 <#FLOB:D01\><h_><p_>4. AUDIENCE: SITUATION AND
CIRCUMSTANCES<p/><h/>
D01   2 <p_>One of the general weaknesses of Bultmann's theology is that
D01   3 its austere challenge is issued directly to each individual,
poor,
D01   4 bare, forked, human animal in isolation from all the rest. It
has,
```

Example from `FLOB_D.txt`

Since the FLOB corpus uses almost the same annotation scheme as the LOB corpus, there is no need to alter the PERL module used above. In this case we simply use the same module but rename it to `flob.pm`. The resulting text output looks like this:

```
4. AUDIENCE: SITUATION AND CIRCUMSTANCES
One of the general weaknesses of Bultmann's theology is that
its austere challenge is issued directly to each individual, poor,
bare, forked, human animal in isolation from all the rest. It has,
```

4.2.6. The COLT Corpus

The COLT corpus (sample input):

```
<REG> S
% reference number // title
<REF> B132401
<TIT> ?
% date // time
<DAT> ?
<TIM> ?
% recording // input device
<DEV> wlk
% duration of conversation
<DUR> 3.26
…
…
<u who=1-1 id=1> Soon find out, cos you do it all the morning and stuff.
 Have you started it?
<u who=1-2 id=2> Yeah.
<u who=1-1 id=3> Oh! Don't. See that was so easy. You don't need to work
it
 out you just sit
<u who=1-2 id=4> That's alright.
```

Example from `B132401.txt`

The COLT corpus has a unique annotation scheme in which every transcript of conversations is introduced by essential information about the circumstances of the dialogs. This comprises e.g. the city, the exact location and setting, used recording devices and the duration of the conversation.
The conversation itself is marked by the beginning tag <u who=…> and can contain further annotations (e.g. noise during conversation) that are enclosed in brackets. Hence the according 'stripping' module has to be more sophisticated than the ones above:

```
sub colt {
        $text_began=0;
        while (<>) {
                chop($_);
                if($_ =~ /^\<u who=\d*\-\d* id=\d*> (.*)$/o) {
                        $text_began=1;
                        push(@lines, $1);
                } else {
                        if ($text_began == 1) {
                                push(@lines, $_);
                        }
                }
        }
        $text = join "\n", @lines;
        $text =~ s/<.*?>//sg;
        $text =~ s/\[|\]//sg;
        $text =~ s/\{|\}//sg;
        print $text;
}
1;
```

The COLT module `colt.pm`

When applied to the COLT text file, the algorithm produces the following output:

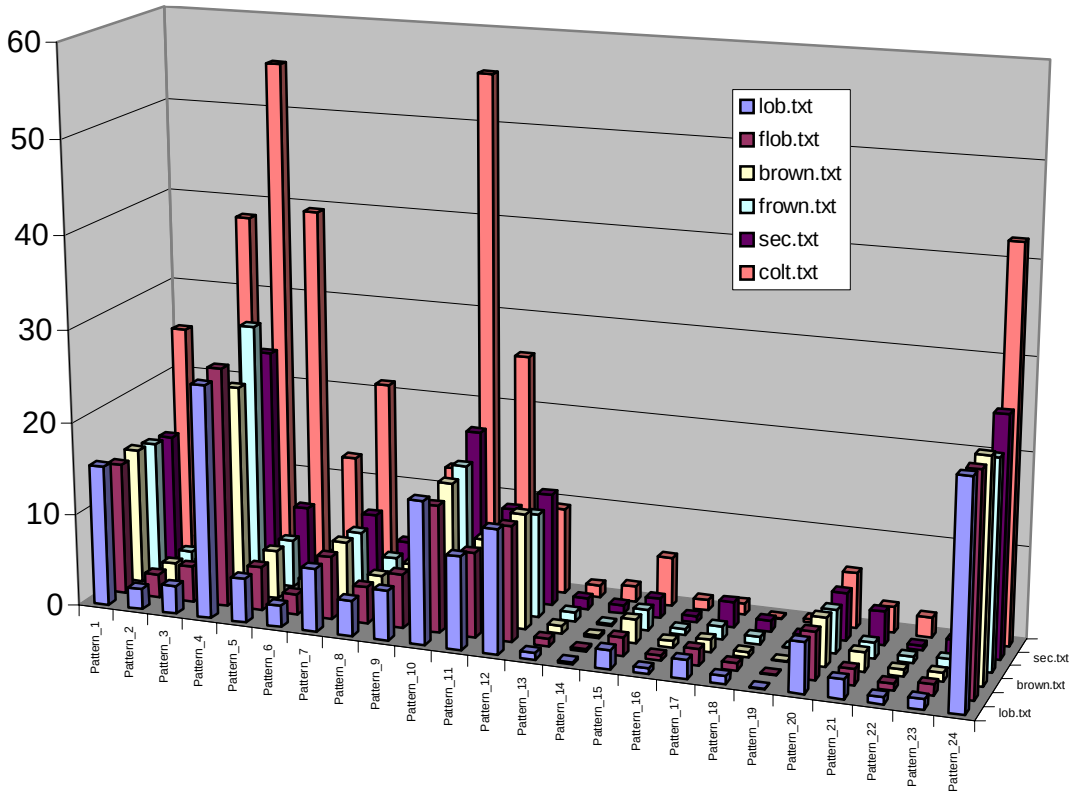
```
Soon find out, cos you do it all the morning and stuff.
 Have you started it?
Yeah.
Oh! Don't. See that was so easy. You don't need to work it
 out you just sit
That's alright.
```

5. Interpretation of Findings

Having processed the raw corpus data by applying the developed system, we are now provided with a wealth of information on the pattern distribution across the different corpora. Since it would be beyond the scope of this study, only brief observations on the generated data can be given.

5.1. General Overview

Figure 1: Occurrences of patterns marking *involvement*[16] (per 1000 words)

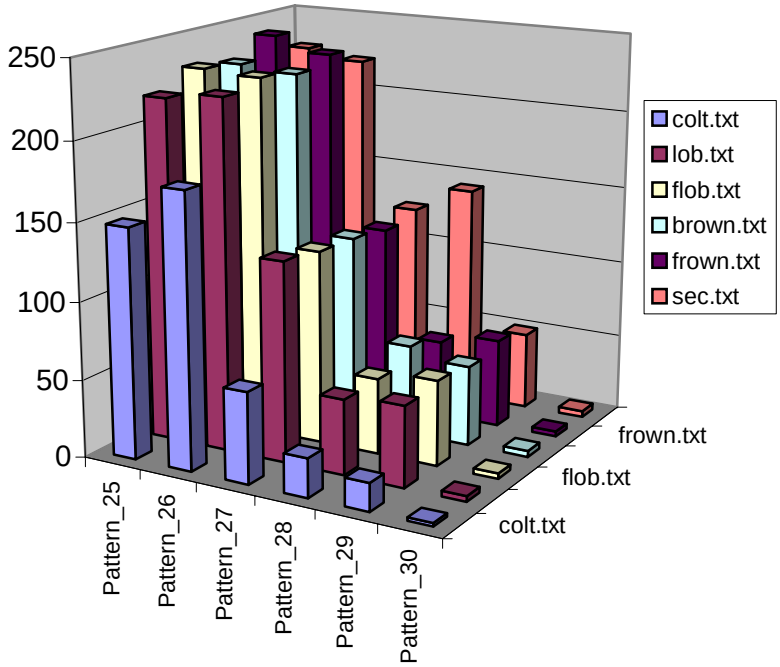


Even a short glance at the chart illustrates the prominent position of the COLT corpus. The higher level of involvement can be explained by the fact that the corpus was composed by gathering transcripts of conversations between London teenagers. We will take a closer look on the most prominent features of the COLT later in this section. Also being a corpus of spoken English, it is surprising that the SEC corpus ranks at almost the same level as the 'traditional' corpora of spoken

[16] Occurrences of pattern 24 (attributive adverbs) were divided by 2 for better viewing.

English, like the LOB and the BROWN corpus. This fact has to be attributed to the composition of the SEC by the Speech Research Group at the IBM UK Scientific Centre, whose primary aim was to collect "samples of natural spoken British English which could be used as a database for analysis and for testing the intonation assignment programs."[17] Comparing the two corpora in raw text format shows that IBM seemingly favored to gather radio news and read stories rather than paying attention to online-produced language such as conversations and dialogues. While this approach is legitimate to analyze intonation mostly, it is still striking that there are no deviations of pattern occurrences from the written corpora as obvious as in the COLT corpus.

Figure 2: Occurrences of patterns marking *information*[18] (per 1000 words)



Taking a look at Figure 2, which shows patterns that occur in complementary distribution to patterns 1-24, we notice that the COLT corpus sports a considerably lower informational content than the written corpora. It is again surprising that the SEC corpus ranks at about the same level of information as the written corpora. What is even more striking is that the SEC has a type/token ratio of 141.75 different lexical items per 1000 words, which marks a high level of information, since the type/token ratio of the written corpora only ranges between 48.35 (LOB)

[17] http://khnt.hit.uib.no/icame/manuals/sec/intro.htm
[18] Occurrences of pattern 26 (mean word length) were divided by 20 for better viewing.

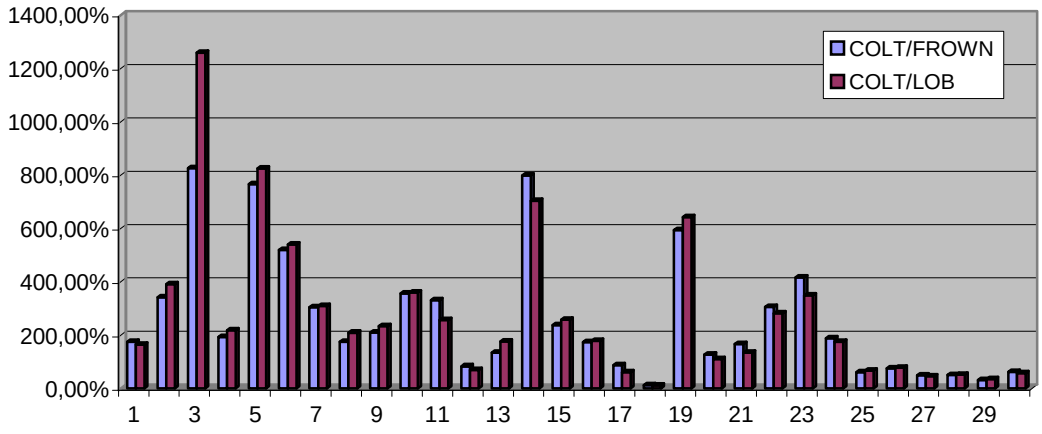
and 58.38 (BROWN).

Having these figures in mind, we can say that the COLT corpus can be regarded as a typical representative of a ‘spoken’ corpus, in a sense that it is composed of conversations in real-time rather than deliberately selecting monologues to be used for intonation studies as found in the SEC. In the next section we will therefore compare the COLT with written corpora.

5.2. Spoken vs. Written Corpora

For a comparison of spoken corpora with their written counterpart, we choose the most typical representatives of their kinds. As Figures 1 and 2 show, the COLT corpus is a very good example for a spoken corpus. Since the written corpora do not differ vastly in occurrence counts, we choose two written corpora to compute a percentile relation of the COLT pattern frequencies. To find statistically relevant patterns we pick two corpora whose contents are as different as possible. Theses are the ‘classic’ LOB corpus (BrE, 1961) and the relatively new FROWN corpus (AmE, 1992). Figure 3 shows the percentage of occurrences of patterns in the COLT in relation to the LOB and FROWN corpora:

Figure 3: Pattern occurrence ratios



As we see, there are many patters that occur up to 12.6 times as often in the COLT as in the written corpora. For an explanation we take the four most

prominent patterns of both figures, which surprisingly are the same in both the FROWN and LOB corpus. These are contractions, 2nd person pronouns, discourse particles, and WH questions. The following table shows the number of occurrences per 1000 words in the respective corpora:

	contractions	2nd person pronouns	discourse particles	WH questions
LOB	3.028	4.776	0.237	0.061
FROWN	4.615	5.141	0.209	0.066
COLT	38.132	39.392	1.670	0.392

Based on their own corpus study, the LGSWE (1995: 166) observes about *not*-contraction that "[f]ull forms are virtually the only choice in academic prose." This explains the lower level of contraction in the written corpora, since they include academic prose as a separate category in their composition. The LGSWE adds that contracted forms of *not* are the most common type of structural reduction apart from contractions of *be*. Therefore, they can be regarded as being a typical representative of contractions. The LGSWE states that contraction is "most common in conversation, followed by fiction and news." (1995: 166). With a striking 3.8%, contractions in the COLT occur 12.6 times as often as in the LOB corpus.

Unfortunately, the LGSWE does not give a numerical distribution of all possible contractions over spoken and written corpora, but it gives an informative table of the proportional use of verb contraction as opposed to *not* contraction (1995: 1132), which confirms the observations made in figure 3.

Considering table 4.33 in the LGSWE (1995: 344), we can explain the high frequency of 2nd person pronouns, which obviously occur in conversation as often as they occur in fiction, news, and academic prose taken together. Regarding figure 4.8 (ibid.), it is mentioned that "[f]irst and second person pronouns, referring to the speaker and the addressee, are naturally very common in conversation because both participants are in immediate contact, and the interaction typically focuses on matters of immediate concern." As the figure shows, personal pronouns account for over 14% of the conversation corpus. In the COLT, there are

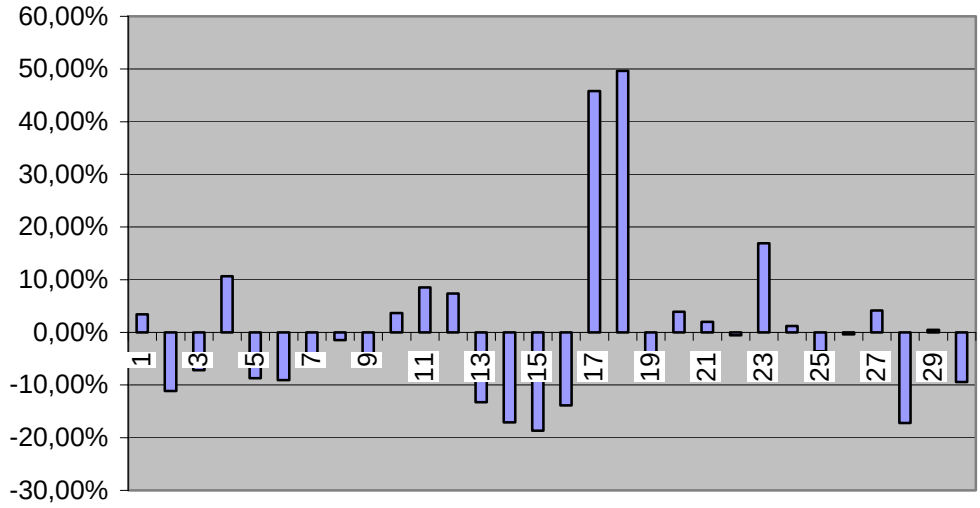
3.9% 2nd person pronouns and 5.6% 1st person pronouns. Comparing these figures to the LOB corpus where we find 0.5% 2nd person and 1.5% first person pronouns, we can call the personal pronouns as being typical for conversation.
Discourse particles occur in the COLT more than 7 times as often as in the written LOB corpus. The LGSWE does not even attempt to establish a table comparing their distribution across conversation and written corpora. Other than that, they compare the distribution of *inserts* across AmE and BrE conversation (LGSWE: 1096).
Other than Biber's algorithm to identify *wh*-questions (cf. 3.19), the LGSWE (1995: 211) simply counts all question marks of a corpus and finds that "[q]uestions are many times more common in conversation than in writing." The table on the same page reveals that the relation of occurrences in conversation to occurrences in written corpora (FICT+NEWS+ACAD) is in fact 47:16=2.9. Compared with the same ratio in COLT/LOB, which is 6.4, we can tell that a high frequency of *wh*-questions clearly mark the corpus as having a high level of involvement.

The prominent peaks in figure 3 can doubtlessly be used to identify a high level of involvement. More than this, one can tell that if an input text shows the same peaks, it also belongs to spoken rather than written English. In section 7, I will give an outlook on how this feature could be used in practice.

5.3. BrE vs. AmE Corpora

For a brief comparison of BrE with AmE, we will use the LOB and BROWN corpora. Figure 4 is calculated as follows: the percentile occurrences of patterns in the LOB is divided by the according figures in BROWN and multiplied by 100 to yield percentages. This results in figures telling that a certain pattern occurs at a rate of x percent in the LOB as it does in the BROWN. To clarify the differences between the two corpora, we subtract 100 from that percentage:

Figure 4: Percentile deviation of pattern frequency (LOB vs. BROWN)



As with the comparison of spoken vs. written corpora, we will again discuss the four most prominent deviations in pattern frequency. In this case, these are amplifiers, sentence relatives on the positive axis. On the negative axis, the most prominent patterns are indefinite pronouns and the type/token ratio (as this pattern occurs in complementary distribution to patterns 1-24, it denotes a decrease in *information*).

Comparing BrE to AmE conversation, the LGSWE (1995: 565) finds that the most common amplifiers (*very, so, really, too,* etc.) occur in almost equal distribution. This notion is obviously far from relating to our findings in the LOB/BROWN comparison, where amplifiers occur 45.8% more often in the BrE corpus. This has to be attributed to the incomparability of conversation and written language and the differing composition of 'amplifiers' (cf. 3.17.).

On sentence relatives, the LGSWE (1995: 609) states that the "[r]elativizers *which* and *that* are the most common overall, but they have notably different distributions across registers […]". Moreover, the LGSWE does not give any information about the distribution of relativizers across BrE and AmE. Still, it is striking that there are 49.6% more sentence relatives in the BrE corpus than in AmE corpus.

As with sentence relatives, the LGSWE does not give information on the distribution of indefinite pronouns across BrE and AmE. The only observation they give is the fact that "[p]ronouns ending in *–body* are more common in AmE than in BrE […]." (1995: 353). As figure 4 illustrates, there are 18.7% less indefinite

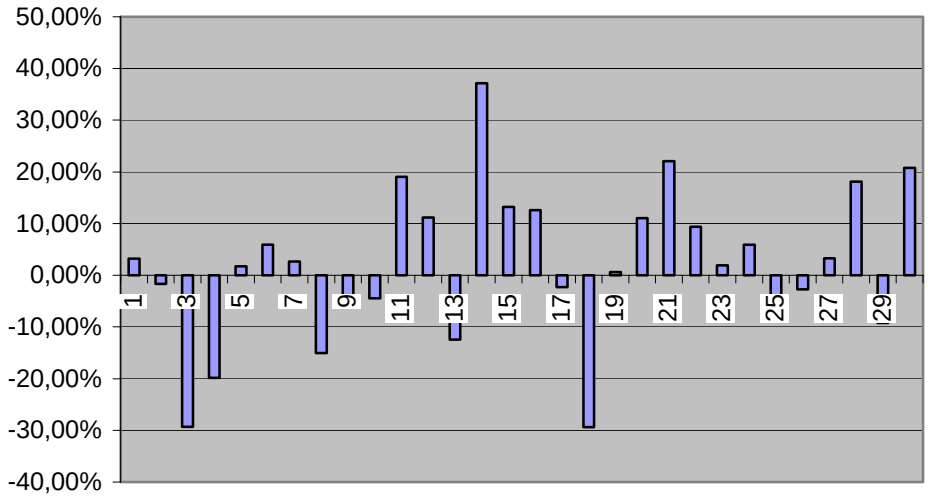
pronouns in the LOB than in the BROWN corpus.

Defining the type/token ratio as "[…] increasing the semantic precision and informational density of a written text […]", the LGSWE (1995: 43) leaves us with the surprising fact that the lexical variation in the BrE corpus is in fact 17.2% lower than in its AmE counterpart.

5.4. A Diachronic Comparison

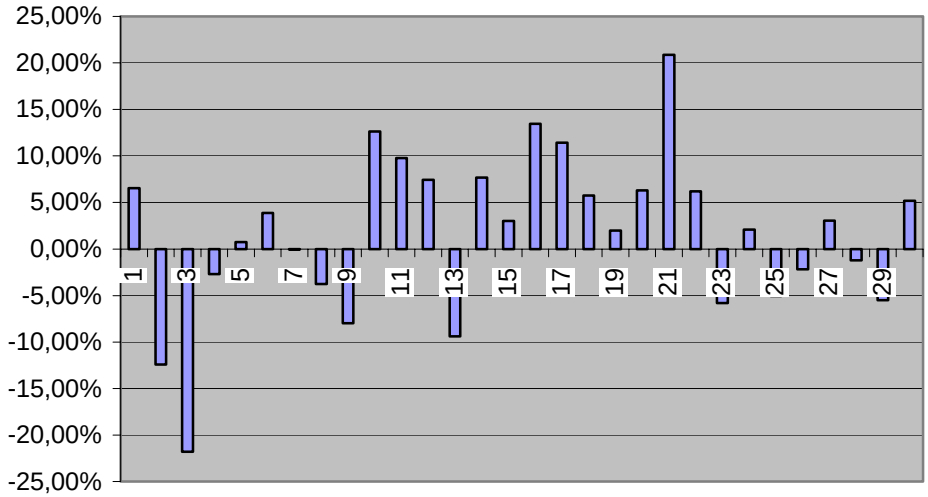
Until now, we have compared pattern frequencies only from a synchronic point of view. In order to obtain an outline of the change of informational and involved content over time, we compute two figures for the LOB/FLOB and BROWN/FROWN corpus pairs:

Figure 5: Percentile deviation of pattern frequency (BROWN vs. FROWN)



The four most prominent features in figure 5 are discourse particles (+37,2%) and nonphrasal coordinations (+22.1%) on the positive axis. Sentence relatives (-29.4%) and contractions (-29.3%) are the most striking deviations on the negative axis. Looking at patterns 1-24, we observe an overall increase of patterns denoting *involvement*. Also there are noticeable deviations in patterns 25-30, denoting *information*.

Figure 6: Percentile deviation of pattern frequency (LOB vs. FLOB)



The four most prominent features in figure 6 are phrasal coordinations (+20.9%) and general hedges (+13.5%) on the positive axis. Contractions (–21,9%) and *that*-deletions (–12.4%) are the most decreasing pattern occurrences when comparing the older LOB corpus with the relatively new FLOB corpus. It seems there is an overall increase on the side of patterns denoting *involvement*, whereas the pattern occurrences that account for informational value of a corpus seem to be almost in balance.

Since the LGSWE does not deal with diachronic analyses, there is no way to compare these findings. As we can see, the generated data provides linguists with large amounts of data to be analyzed. I will give possible applications of the data in section 7.

6. Problems

In the course of developing the system on hand, I encountered problems not so much on the technical side but rather in the theoretical framework found in Biber (1988). One big problem is that Biber's definitions of patterns often differ greatly from the definitions found in standard grammars like Quirk (1985). For example, Biber's notion of *general emphatics* (cf. 3.9.) is an arbitrary mixture of various grammatical phenomena like e.g. emphasizers (Quirk 1985: 583), boosters (Quirk

1985: 591), the intensifying *such* (LGSWE: 282), the auxiliary *do* in emphatic function (LGSWE: 433) or the periphrastic comparison with *more, most*. Since it is not clear where Biber takes these definitions from, I had to copy Biber's pattern as faithful as possible from the rather vague definitions he gives in Biber (1988:224-245).

Having done this, I was often confronted with outputs that did in many cases not comply with the respective definitions. That was the case e.g. in pattern 3.2. (*that*-deletion):

> …scientists find Don Cupitt unscientific…
> …to see this poll as a…

These examples of matches for *that*-deletion (from the SEC corpus) show that either Biber's algorithms were inaccurately defined or there was an inaccuracy in the involved tagging process.

Also Biber's composition of different cooccurrence patterns sounds rather far-fetched if we look at statements like:

> "…widest possible range of *potentially* important linguistic features…"
> "…*might* be used to differing extents in different types of text…"
>
> (Biber 1988: 72)

A big obstacle for the tagging process is inaccurate sentence punctuation in the input files. Many patterns rely on correct punctuation as well as the TaggerTagger. Since TreeTagger employs probabilistic methods to assign tags to the input files, it can only produce accurate tags if the input 'looks familiar', which means that the input should look like the texts that TreeTagger was trained with. If this is not the case, the resulting tags can be very inaccurate and therefore bias the subsequent parsing of cooccurrence patterns, which again results in heavily skewed CSV files. Especially in the case of the LLC corpus, where there was no viable way to reconstruct the sentence sequences by assigning accurate sentence punctuation, the resulting tagged files also looked very inaccurate and could simply not be used in the present study. Although there is sentence punctuation in the COLT corpus, the relatively short sentences could have caused the tagger a hard time, but TreeTaggers output still looked quite accurate.

For the brief overview of the resulting CSV files, the LGSWE (1995) was not as supportive as I had expected since there is not consequent analysis of e.g. AmE/BrE and CONV/written distributions in the LGSW corpus. Other than that, the composition of the distribution tables looks utterly arbitrary, which does not allow for a direct comparison of my findings with previous study.

7. Conclusion and Outlook

As opposed to studies that focused on the distribution of single grammatical phenomena, Biber's multi approach provides a new, statistically sound basis for text type categorization. Build on a wide variety of relevant features, Biber's 'dimensions' will be the a starting point for automatic text type analysis.

Despite the aforementioned problems, the system at hand still provides a unique method of identifying the level of *involvement* and *information* in texts. The complete automation of the system allows for unattended text type analysis which can be used to categorize texts from various sources. For example the system could analyze the informational content of websites, which can be quite convenient in an era of quickly increasing information at decreasing spare time. In this manner, one can tell interesting sites from the uninformative rest.

As a basis for other linguists, a dedicated web server could be set up that accepts raw text delivered by email, which then analyzes the input and sends back a complete CSV file holding all data on pattern occurrences in the text. As a variation, the system could be set up as a CGI supported website, that analyzes either text entered into a form or crawls though a given URL returning to the user a profile of the informational content of the desired site.

As a further option, optical character recognition (OCR) could be extended to something like 'optical text type recognition'. If the pattern distributions of e.g. tabloids and weekly journals are known to the system, it can try to identify the respective newspaper, telling if it is a tabloid or a weekly journal.

The vast amount of calculated data could provide topics for dozens of other studies (e.g. analyses of the different categories found in most corpora). Moreover, the results can be used as input for the calculation of 'dimension scores', which are an essential figure when it comes to comparing the "values of features" (cf.

Biber 1988: 94). Since this statistic procedure maps results as produced by my system to a one-dimensional scale, it allows for easier comparison of text types.

Finally, the whole system could be altered to analyze other languages than English, providing a general tool for processing texts of any sort imaginable.

8. References

BIBER, DOUGLAS. Variation across speech and writing. Cambridge University Press, Cambridge, 1988

BIBER, DOUGLAS. A typology of English texts. In *Linguistics 27*. Mouton de Gruyter, Amsterdam, 1989

BIBER, DOUGLAS ET AL. Longman grammar of spoken and written English. Pearson Education Limited, 1999

HILLER, ULRICH. "Contracted Forms im gesprochenen Englisch: ihre Frequenz und Distribution als Funktion des Sprachregisters." In *Die Neueren Sprachen* 82: 15-27. 1983

MASON, OLIVER. Programming for Corpus Linguistics: *How to do text analysis with Java*. Edinburgh University Press, Edinburgh, 2000

TOTTIE, GUNNEL. Negation in English speech and writing : *a study in variation*. Academic Press, Inc. , San Diego, 1991

QUIRK, RANDOLPH ET AL. A comprehensive grammar of the English language. Longman Group Limited, 1985

Software

Phrasys Java text tokenizer:
<http://www.phrasys.com/software/dist/003/index.html>

QTAG POS tagger:
<http://www.english.bham.ac.uk/staff/oliver/software/tagger>

PERL Programming Language:
<http://www.perl.com>

TreeTagger POS tagger:
<http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html>

RedHat Linux operating system:
<http://www.redhat.com>

Hiermit bestätige Ich, dass Ich diese Magisterarbeit selbständig verfasst und keine anderen als die von mir angegebenen Hilfsmittel benutzt habe.

_____

Marc Reymann

Regensburg, den 7. Juni 2002