BNC*web* (CQP-edition): The marriage of two corpus tools

Sebastian Hoffmann, University of Zurich

Stefan Evert, University of Osnabrück

Abstract

In order to realize the full potential of electronic corpora, most of today's linguists depend on the availability of specialized software tools. In this paper, we first present two existing software packages – BNC*web* and the CQP query processor – and discuss their strengths and drawbacks. We then show that a marriage of these tools has led to a new package that combines the efficiency and flexibility of CQP queries with the user-friendliness of BNC*web* and its wide range of post-query features. Finally, we outline a blueprint for a more general search tool which we plan to implement in the near future.

1 Introduction

Since the release of the Brown corpus in the early 1960s, the number of electronic corpora available for linguistic research has grown steadily. As a result, the corpus linguist of today can choose from a large pool of both general and specialized collections of authentic language data to suit his or her research requirements.[1] In view of their size and often complex data structures, access to electronic corpora typically requires the use of specialized corpus tools that allow the researcher to conduct fast and reliable searches over large amounts of text. In addition, many corpus tools standardly offer a whole range of post-query features for further analysis of the retrieved data (e.g. relative frequency counts, calculations of collocational strength, distribution of query results over various textual categories, etc.).

Given this reliance on tools that mediate between human researcher and electronic text, the feasibility of linguistic investigations depends at least partly on the quality and the range of features offered by the software. Developers of such software tools thus shoulder a considerable responsibility and face a tightrope walk between providing maximum ease of use and offering the greatest possible flexibility of searches and analyses.

[1] At the time of writing, the most complete and up-to-date list of available corpora was David Lee's Web page at <http://devoted.to/corpora> (20 November 2005).

In the present paper, we will first introduce readers to two rather different corpus tools that have been available to linguists for some years, viz. BNC*web*, a Web-based interface to the 100-million word British National Corpus (BNC), and the IMS Corpus Workbench (CWB), a generic query engine for large text corpora that was developed for applications in computational lexicography. After a description of the design as well as an overview of the strengths and weaknesses of each software package, we will briefly discuss the characteristics of an ideal corpus tool. As a first step towards the creation of such a tool, we will demonstrate how BNC*web* and the Corpus Workbench can be combined to provide a more powerful gateway to the BNC – without at the same time having to cut back on aspects of user-friendliness, performance or flexibility. Two practical examples will be given to illustrate some of the advantages gained by the marriage of these tools. In the final part of this paper, we will then look to the future and present an outline for the development of a more generic corpus tool which will be closely modelled on the structure and functionality of BNC*web* but allow researchers to apply its extensive range of features to any text corpus that is made available in a suitable format.

2 BNC*web*

BNC*web* is a Web-based client which allows users to access the BNC and its rich levels of metatextual annotation by means of a standard Web browser. It was developed for internal use at the University of Zurich (Lehmann et al. 2000) but was subsequently released to the general public (non-commercial use only) in the year 2002. It is distributed in its current version 2.1 for a nominal fee. BNC*web* relies on the SARA server software (sarad, included with the official BNC distribution) to make indexed searches via a simple and user-friendly interface. Results are presented in KWIC (key word in context) view or as a list of entire sentences. Links are provided for users to access the larger context of individual matches as well as the relevant bibliographical and speaker information, if available. Like many other corpus tools, BNC*web* offers a range of additional features for the analysis of the retrieved data. These include, for example, the display of sorted search results, collocations, frequency distributions, random thinning of query results and the manual deletion of individual hits. A query history provides users with quick access to previous searches conducted with BNC*web* and post-processed concordances can be saved and easily retrieved for further analysis. In addition, search results can of course also be saved to the user's hard disk.[2]

[2] A comprehensive description of all features of BNC*web* is available on-line at <http://homepage.mac.com/bncweb/> (20 November 2005). Readers may also be interested in the critical evaluation presented in Kreyer and Mukherjee (2002).

BNC*web* is implemented using a standard client-server architecture. It consists of a set of CGI-scripts written in Perl that are invoked by a Web server (typically Apache). The scripts interact with the SARA server software and format its output as HTML documents that are sent to a Web browser running on the client computer. Query results are also internally stored in MySQL database tables, which form the basis for many of the core post-query features of BNC*web* (e.g. collocations, distribution, sort, etc.). All necessary components of the system thus reside on a server and no proprietary software is required on the part of the end-user.

While BNC*web* clearly represents an attractive option for accessing the wealth of data contained in the BNC, it also has some obvious drawbacks, both in terms of its feature set and in terms of its technical implementation. Since BNC*web* is an interface and not an independent client, it necessarily inherits the limitations of the SARA server software on which it is based. For example, searches must be lexical and thus cannot involve grammatical patterns which are defined by sequences of part-of-speech tags.[3] On a more general level, since BNC*web* has been developed exclusively for the BNC, it consequently cannot be easily adapted to work with other corpora – even if they were available in a similar format. Finally, although BNC*web* requires no specific software on the client computer, the fact that its components have to be installed on a Unix server no doubt represents a considerable hurdle for anyone who is not familiar with at least some level of system administration.[4]

3 The IMS Corpus Workbench

The IMS Corpus Workbench (CWB, Christ 1994) is a software package designed to process large text corpora of 100 million words and more. The Corpus Workbench has been developed at the University of Stuttgart since 1993, and version 3.0 has recently been released as open-source software under the GPL license.[5] Originally designed for applications in computational lexicography, the CWB focuses on word-level annotations such as parts of speech, lemmatization and morphological features, but there is also some support for document metadata and structural markup (such as noun chunks or

[3] This limitation is determined by the fact that the index which is accessed via the sarad server does not contain a layer for part-of-speech tags.

[4] BNC*web* requires access to a full installation of the BNC World Edition (with its index files and server software). In addition, the relational database MySQL and some Perl modules may need to be installed. Although a manual is provided, previous experience with Unix system administration is an advantage.

[5] See the CWB homepage <http://cwb.sourceforge.net/> (6 December 2005) for more information on the software and how to obtain it.

multi-word expressions). CWB corpora are fully indexed and stored in a compact binary format, which permits efficient searches and data retrieval. There is no easy way of modifying the encoded text and its annotations, but new, independent annotation layers can be added at any time.

The central component of the Corpus Workbench is the corpus query processor CQP. Its query language allows sophisticated searches both for individual words (which can be matched against regular expressions) and for lexico-grammatical patterns (using linear grammars that have access to all levels of annotation).[6] When CQP is utilized as a command-line tool, the results of a corpus query are displayed one screen page at a time in a terminal window, using a customizable KWIC format. Interactive commands allow users to sort, filter and merge query results, save them to the hard disk, and compute simple frequency lists. The query processor can also be operated in batch mode, e.g. as one component of a system for the automatic extraction of corpus frequency data.

The particular strengths of CQP are (i) the integration of an unlimited number of word-level annotations, document metadata and structural markup (in the form of XML start and end tags) in its queries; and (ii) the ability to perform very general searches (e.g. purely grammatical patterns such as noun phrases) on large corpora and efficiently handle the millions of hits they may return. A simple macro expansion mechanism allows complex queries to be broken down into manageable parts, which can be stored in macro libraries for later re-use. Based on the CQP macro language, Evert and Kermes (2003) have implemented a broad-coverage shallow syntactic parser for German and a system for the automatic extraction of various types of subcategorization information.

The speed and flexibility of the Corpus Workbench come at a price that has to be paid mostly by inexperienced users and those without strong computer skills. Consequently, the Corpus Workbench has so far mostly been used in computational linguistics and corpus linguistics departments where help from a local expert, usually an experienced software developer, is readily available. The most problematic aspect of CQP is certainly its complicated query language, which even advanced users find difficult to memorize in all its details. There is also a relative dearth of post-query features such as the frequency distribution and collocation analysis offered by BNC*web*. While it is possible to sort query matches and calculate frequency tables directly in CQP, additional processing with Perl scripts or other external tools is usually required in order to present the results in an accessible manner (cf. Section 6). Finally, the Corpus Workbench suffers from the lack of a user-friendly graphical interface to the query processor, which is only available as a command-line application. Corpus

[6] A full description can be found in the CQP query language tutorial, which is available on-line from <http:/cwb.sourceforge.net/documentation.html> (6 December 2005).

queries as well as interactive functions have to be formulated as complex and often unintuitive commands. The results are displayed one screen page at a time, but there is no easy way to jump to a specific page in this KWIC display or to access additional context for a single match.

4 The ideal corpus tool – or "squaring the circle"

In an ideal world, linguists would have at their disposal a corpus search tool that combines all the desirable characteristics listed below:

- full flexibility of corpus searches; neither the query language nor the user interface impose any restrictions on the complexity of searches;
- highly intuitive and user-friendly query specification; even novices face no difficulties in conducting searches of a very complex nature, and simple word or phrase queries are possible without consulting the query language documentation;
- high speed; even very large corpora (of 100 million running words and more) can be searched quickly and efficiently;
- no restriction to a specific corpus, corpus format or size; new corpora can be integrated into the system with only a minimal need for manual intervention and configuration; all levels of annotation are automatically recognized and integrated into the full set of post-query features available;
- unlimited range of annotations supported by the query language (e.g. word forms, lemmas, part-of-speech tags and other token-level annotations, text-level and utterance-level metadata, recursive syntactic analyses, etc.);
- ability to work with large numbers of matches in the post-query features;
- flexible and intuitive display of query results (with extended context); the complete set of corpus annotations relevant for each match can be conveniently accessed;
- extensive range of flexible post-query features (sorting, frequency tables, collocation analyses, etc.); computer-savvy linguists should be able to create their own modules for post-query processing;
- possibility to add new user-defined annotation levels to the corpus (e.g. annotation of pragmatic features); user-defined and original annotation levels can be freely combined in searches and post-query features;
- manual categorization of query results; users can analyze and annotate individual query matches and compute statistical analyses of their annotations; again, all levels of the original corpus annotation remain available and can be freely combined with user-defined categories;

- platform-independent off-the-shelf solution that comes with an installation script and requires neither programming nor system administration skills;
- stable and robust implementation that does not crash even on faulty input and supports a large number of concurrent queries without exhausting the server computer's resources (such as memory, disk space and CPU time).

It is of course not possible to combine all these strengths in a single tool. Instead, trade-offs are necessary, either because different requirements are mutually incompatible (e.g. full flexibility of searches paired with complete ease of use) or simply because they cannot all be satisfied with the limited time and manpower typically available for the development of corpus software. Existing tools put the emphasis on some of the criteria and compromise in other respects, as we have seen for BNC*web* and the Corpus Workbench. For example, while BNC*web* offers user-friendliness and an impressive range of post-query features, it is restricted with respect to the flexibility of searches that can be conducted. Conversely, the Corpus Workbench excels in the expressiveness and versatility of its query language but sorely lacks an intuitive graphical user interface.

Naturally, developers of corpus tools should nevertheless strive to square the circle and satisfy as many items as possible from the list of characteristics shown above. Since many limitations are simply due to the limited resources available for software development, it is to be hoped that a considerable step in this direction might be achieved by combining existing tools whose strengths (i.e. the aspects on which their implementation has focused) are complementary. As we have shown in the previous sections, BNC*web* and the Corpus Workbench are two such tools. The goal of their marriage, which is described in the following, was to merge complementary strengths without introducing new weaknesses at the same time. In Section 5, we offer a brief description and evaluation of this process, followed by a presentation of some features of the newly created tool in Section 6.

5 Combining forces

Since some of the more prominent limitations of BNC*web* are inherited from SARA, it was a logical conclusion to replace the sarad server with an alternative tool that would enable users to conduct more flexible searches over the BNC. Fortunately, there are enough similarities between CQP and SARA to make such a replacement feasible without having to rewrite large portions of the Perl code.

In a first step, the main task consisted in replacing all calls to sarad with their CQP counterparts.[7] Furthermore, since the output format of sarad and the Corpus Workbench are not identical, it was also necessary to make some changes to the way results are handled internally before they are displayed to the user. Finally, differences in the query syntax of SARA and CQP meant that those portions of Perl code had to be adapted which convert the input received from the Web client into well-formed query expressions. While these changes may seem quite substantial at first sight, they were in fact astonishingly easy to implement. It must be noted, however, that a more modular design of the BNC*web* code would certainly have facilitated this undertaking.

Once the full functionality of the system had been re-established, our attention turned to aspects of optimization. In this context, it was a distinct advantage that developers of both original tools were involved in the project so that modifications could be made both to BNC*web* and to the CQP query processor. In some cases, the addition of new functions and output options to CQP dramatically increased the overall speed of the system.[8]

However, the integration of the Corpus Workbench and BNC*web* was not without problems: together with the strengths of CQP, BNC*web* also inherited its complex query language. As a result, even a search for a simple word form or phrase could not be achieved without a relatively complicated query expression.[9] In order to meet our goal of combining strengths without introducing new drawbacks, a simplified query language had to be created that rendered the CQP-edition of BNC*web* as user-friendly and intuitive to operate as the original BNC*web,* thus allowing even novice users to conduct queries of considerable complexity. Readers will obtain a general impression of the functionality of this simplified query language in the following section, which discusses two sample queries in greater detail.

[7] The replacement was also facilitated by the fact that the Corpus Workbench provides a comprehensive and well-documented interface to the Perl programming language, which was easily integrated into the Perl scripts of BNC*web*.

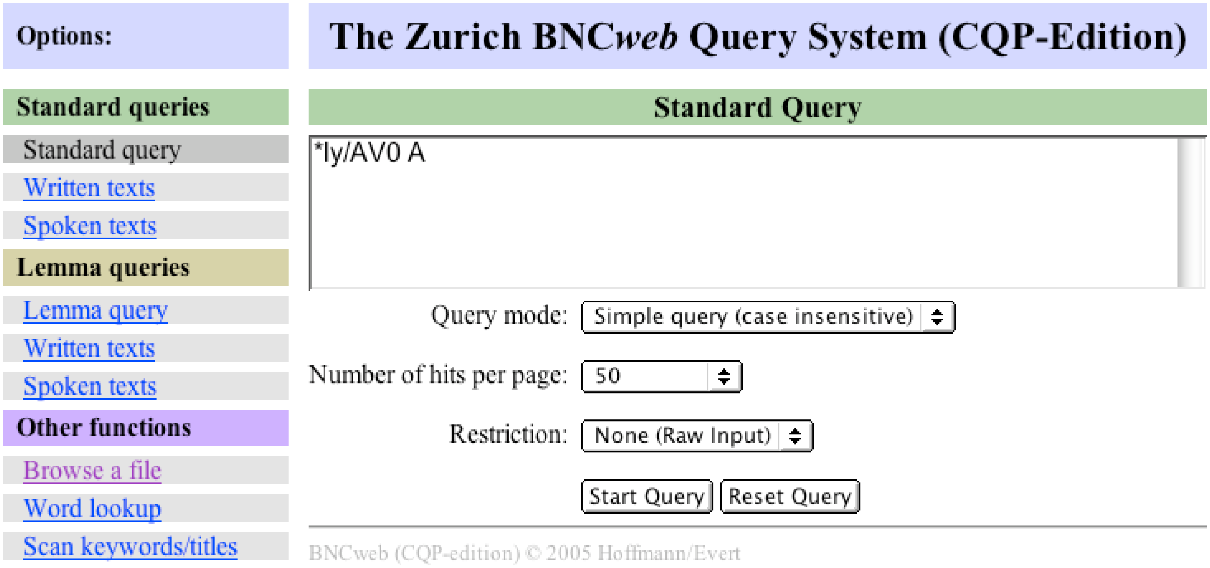
[8] For example, CQP was modified to provide all of the following information at once: the matched string and its immediate context, the corpus positions of the beginning and the end of the matched string, the name of the BNC text, the s-unit number and – if applicable – the speaker identification code. Retrieval of this information had previously required several unrelated queries, thus considerably slowing down the overall performance of some post-query features in BNC*web*.

[9] For instance, a case-insensitive search for the phrase *red herring* had to be performed with the fairly complicated CQP query `"red"%c "herring"%c;`.

6 Two sample queries

In the current section, we will present two practical examples that are intended to demonstrate some of the benefits gained by combining the forces of BNC*web* and CQP. In addition, readers who have never used BNC*web* before may find this section useful to form an impression of the general functionality of this tool.

The first of these examples is concerned with adjective intensification such as *terribly good* or *very high* (cf. for example Lorenz 1999). In the original version of BNC*web*, the lack of part-of-speech information in the index meant that such instances of intensification could only be retrieved if at least one of the two lexical entities was specified in the initial query. For instance, it was possible to look either for all occurrences of *terribly* that are followed by an adjective or for all instances of *good* that are premodified by an adverb. For a comprehensive analysis of the phenomenon of adjective intensification, however, a purely part-of-speech based pattern search – e.g. 'any adverb followed by any adjective' – would of course be much more useful. In the CQP-edition of BNC*web*, such a search no longer presents any difficulties. For the sake of simplicity, we will concentrate in the following on one of the various forms of adjective intensification and only consider instances of adjectives that are immediately preceded by an adverb ending in *-ly*.



The Zurich BNC*web* Query System (CQP-Edition)

Standard Query

*ly/AV0 A

Query mode: Simple query (case insensitive)

Number of hits per page: 50

Restriction: None (Raw Input)

BNCweb (CQP-edition) © 2005 Hoffmann/Evert

Figure 1: *The standard query window in BNC*web *(CQP-edition)*

Figure 1 displays a screen shot of the standard query form of BNC*web*.[10] Query strings can be entered into the large text area and the "Start Query" button will initiate the query and display the query result. By default, the "Simple query" mode is selected, which allows the user to enter search strings in a greatly simplified query format rather than in the much more complex CQP syntax (cf. Section 5).

The query string "*ly/AV0 A" (as shown in Figure 1) contains the wildcard character '*', which is used to match one or several consecutive word-characters (i.e. non-whitespace characters). The expression '*ly' thus retrieves any word – of any length – that ends in *-ly*. The slash attached to this expression indicates that the following characters should be interpreted as a part-of-speech code; in this case, the search is limited to 'AV0', which is the basic tag for adverbs in the BNC tagset.[11]

The second part of this query string is represented by the letter 'A'. By default, the simple query mode interprets a sequence of capital letters as a part-of-speech code even though they are not preceded by a slash.[12] In the case of the single letter 'A' (for 'adjective'), this is a shortcut for the various possible adjective tags (e.g. AJ0, AJC, etc.).

While many indexed searches over the whole 100-million word BNC only take seconds (e.g. searches for a single word or a sequence of lexical items), the query shown in Figure 1 requires a little more patience from the user. For example, with BNC*web* installed on a standard Apple PowerBook laptop computer (G4 processor, 1.5 GHz, 1 GB RAM, Mac OS X 10.4), it takes approximately 55 seconds to conduct this search.[13] Future optimizations of the Corpus Workbench may lead to a reduction of this figure.

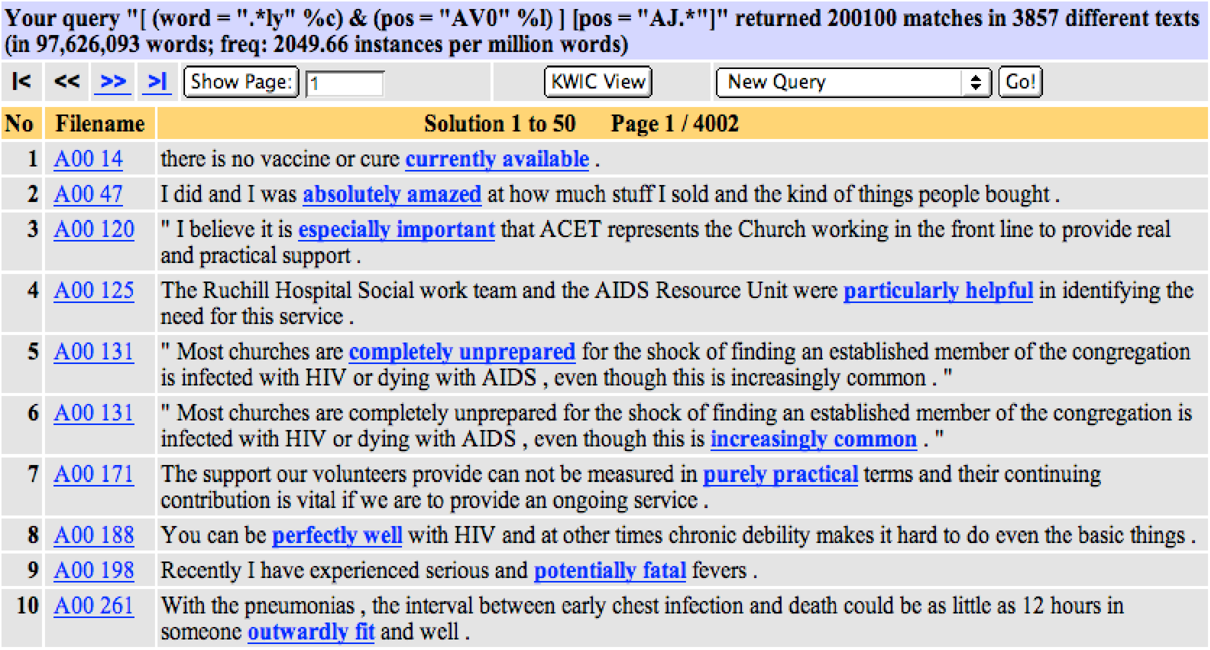
[10] In order to save space, all screen shots in this paper are cropped to include only those areas of the screen which are relevant to the discussion.

[11] Other adverb tags are AVQ (interrogative and relative *wh*-adverbs, e.g. *where*, *when*) and AVP (adverbial particles, e.g. *up* in *give up*). For further information about the BNC tagset, see Leech & Smith (2000).

[12] This convention can lead to unexpected results when users search for abbreviations that only consist of capital letters. For example, the character sequence "US" in the query "the US government" will be interpreted as 'any word with the part-of-speech tag "US"'. Because such a tag does not exist, no matches will be retrieved. Since simple query searches are by default case-insensitive, this situation can be avoided by entering abbreviations in lowercase letters, or by explicitly indicating that "US" refers to the word form: "the US/* government".

[13] All benchmarking information given in this paper will be based on timings taken with the same set-up. The Apple PowerBook is a standard consumer laptop computer with moderate performance levels. Installing BNC*web* on top-of-the-range server hardware will of course result in drastically faster processing.

Figure 2 displays the first ten instances of the query result for the search string "*ly/AV0 A".[14] Within every sentence, the words matched by the search string are underlined and represent a hypertext link to a separate page that displays the individual match in its larger context. A second link is provided to the left of each sentence, which leads to a page containing bibliographical information about the relevant BNC text. Users can navigate through the query result with the help of links located above (and also below) the displayed set of sentences. Since only small sections of the query result are sent to the client computer at a time (the default is 50 sentences), the total number of instances retrieved by a query has no influence on the speed of this operation and users can quickly jump to any page number of their choice.



Your query "[(word = ".*ly" %c) & (pos = "AV0" %l)] [pos = "AJ.*"]" returned 200100 matches in 3857 different texts (in 97,626,093 words; freq: 2049.66 instances per million words)

No	Filename	Solution 1 to 50 Page 1 / 4002
1	A00 14	there is no vaccine or cure **currently available** .
2	A00 47	I did and I was **absolutely amazed** at how much stuff I sold and the kind of things people bought .
3	A00 120	" I believe it is **especially important** that ACET represents the Church working in the front line to provide real and practical support .
4	A00 125	The Ruchill Hospital Social work team and the AIDS Resource Unit were **particularly helpful** in identifying the need for this service .
5	A00 131	" Most churches are **completely unprepared** for the shock of finding an established member of the congregation is infected with HIV or dying with AIDS , even though this is increasingly common . "
6	A00 131	" Most churches are completely unprepared for the shock of finding an established member of the congregation is infected with HIV or dying with AIDS , even though this is **increasingly common** . "
7	A00 171	The support our volunteers provide can not be measured in **purely practical** terms and their continuing contribution is vital if we are to provide an ongoing service .
8	A00 188	You can be **perfectly well** with HIV and at other times chronic debility makes it hard to do even the basic things .
9	A00 198	Recently I have experienced serious and **potentially fatal** fevers .
10	A00 261	With the pneumonias , the interval between early chest infection and death could be as little as 12 hours in someone **outwardly fit** and well .

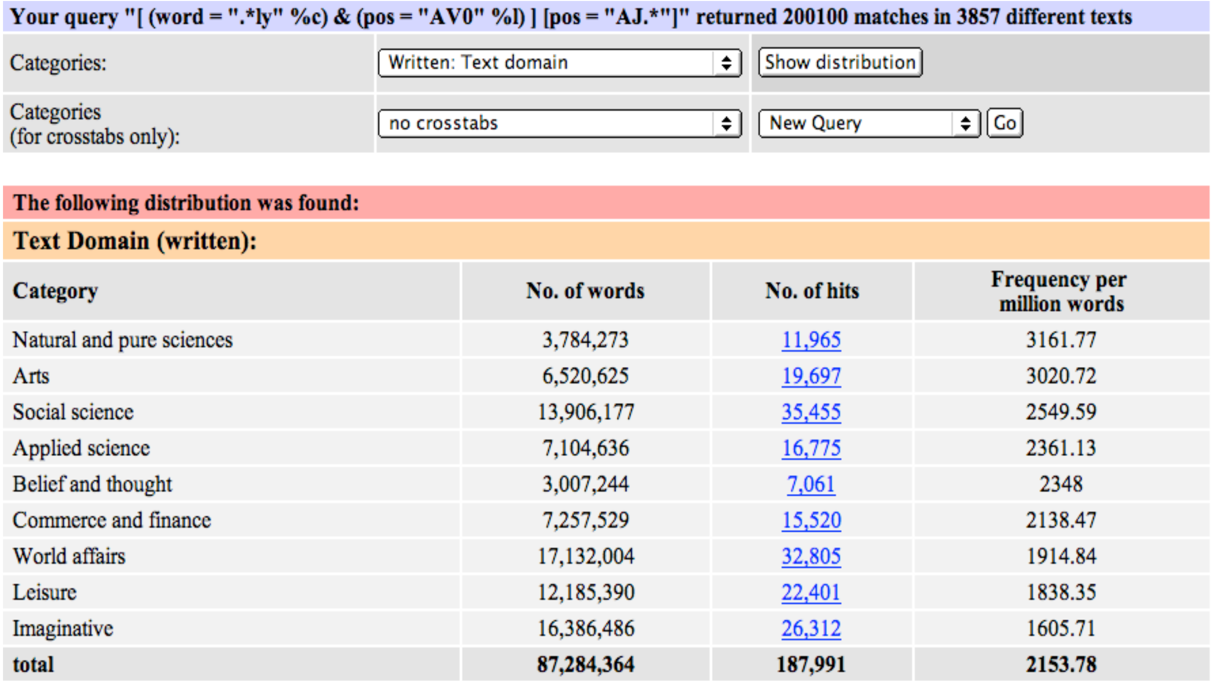
Figure 2: *Result of the simple query for "*ly/AV0 A"*

Figure 2 also shows the title bar, which is positioned at the top of each set of results. In addition to the total number of matches (here 200,100), it also displays the number of different BNC texts in which these matches were found. Such information makes it possible to evaluate the general currency of a word or construction; highly specialized vocabulary or idiosyncratic uses may thus easily

[14] It is worth noting that not all the retrieved instances are examples of adjective intensification, of course (e.g. *currently available* in the first sentence). The precision of tag-based pattern searches is rarely 100 per cent and considerable manual post-processing may often be required to arrive at a completely reliable set of results.

be distinguished from more generally employed linguistic features. More importantly, BNC*web* also presents the user with information about the relative frequency of the query result. This information is also calculated when a search is restricted on the basis of metatextual annotation (e.g. 'Age of author' or 'Medium of text'). Such relative frequency counts provide a convenient yardstick for the comparison of linguistic phenomena across different textual categories and are thus a basic and fundamentally important – but all too often missing – feature of any corpus tool.

Finally, the title bar also displays a translation of the simplified query into the much more complex CQP search string that was used by the BNC*web* scripts to interact with the Corpus Workbench. Users who are interested in learning the CQP query syntax may find this feature useful for acquiring a better understanding of its rules.



Your query "[(word = ".*ly" %c) & (pos = "AV0" %l)] [pos = "AJ.*"]" returned 200100 matches in 3857 different texts

Categories: Written: Text domain — Show distribution

Categories (for crosstabs only): no crosstabs — New Query — Go

The following distribution was found:

Text Domain (written):

Category	No. of words	No. of hits	Frequency per million words
Natural and pure sciences	3,784,273	11,965	3161.77
Arts	6,520,625	19,697	3020.72
Social science	13,906,177	35,455	2549.59
Applied science	7,104,636	16,775	2361.13
Belief and thought	3,007,244	7,061	2348
Commerce and finance	7,257,529	15,520	2138.47
World affairs	17,132,004	32,805	1914.84
Leisure	12,185,390	22,401	1838.35
Imaginative	16,386,486	26,312	1605.71
total	**87,284,364**	**187,991**	**2153.78**

Figure 3: *The distribution of adjective intensification (adverbs ending in* -ly*) over the BNC text domains*

It is beyond the scope of this paper to present a complete overview of the functionality of BNC*web*. For the example of adjective intensification, we would therefore like to restrict ourselves to a brief description of two post-query options. The first of these is the distribution feature. Very often, researchers may glean interesting information about the usage of a linguistic phenomenon by looking at its distribution over different textual categories. This type of descriptive statistics can be conveniently compiled with BNC*web* even if the query result consists of several hundred thousand matches. For example, it takes about 15 seconds to calculate distribution information for the 200,100 instances of (potential) adjective intensification in the BNC. Once this information has been compiled and internally stored in a MySQL database, users can quickly switch between different types of metatextual categories.

Figure 3 shows the distribution of our query result over the nine text domains in the BNC. It reveals that texts classified as belonging to the domain 'Natural and pure sciences' are almost twice as likely to contain adverb-adjective sequences as imaginative prose texts (3,162 per million words vs. 1,606 pmw).

The second feature to be mentioned here is the sort feature which offers a number of options for arranging the sentences of a query result in different order. For instance, users may want to sort all matches alphabetically with respect to the first word that follows the item(s) matched by the query string. In addition to simple sorting, users can also assign a part-of-speech based filter to the sorted position. For example, the sorted result can be restricted to those matches which are immediately followed by a noun. Such a sorted list can be used to detect patterns of use (e.g. common co-occurrences in the immediate context, semantic prosodies, etc.) that might otherwise remain hidden.

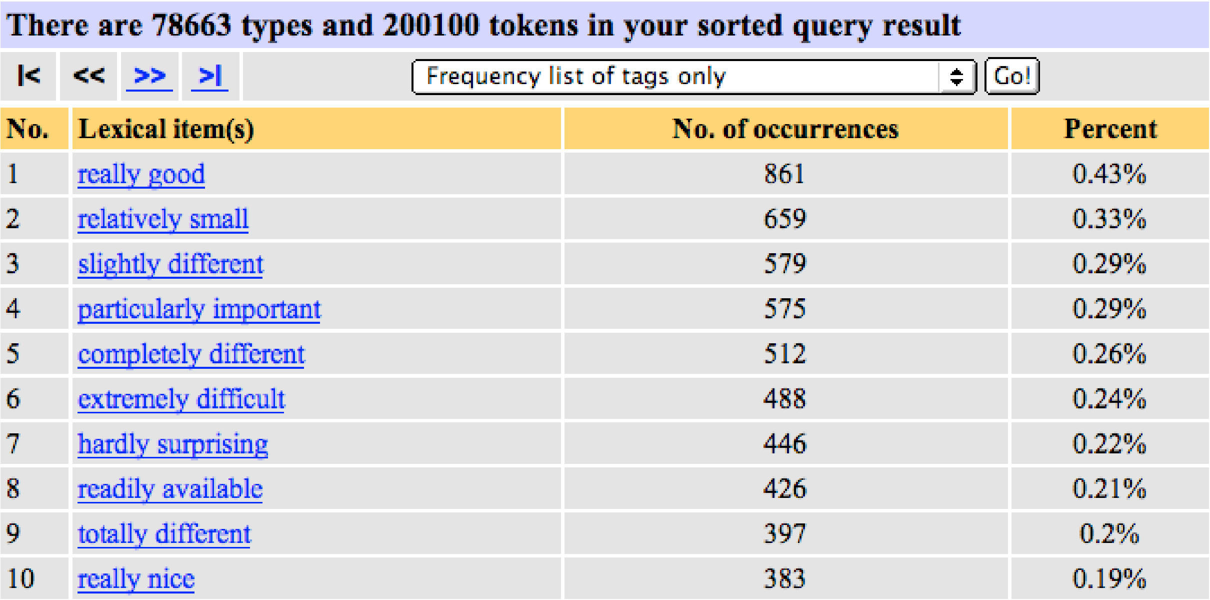
In the context of adjective intensification, researchers may also be interested in finding out which adverb–adjective combinations occur most frequently in the corpus. With BNC*web*, this can be answered by sorting the query result on the node, i.e. on the lexical items matched by the query string.[15] Since sorting a query result requires more data to be processed and stored internally than the compilation of a distribution analysis, it takes considerably longer to sort the complete set of 200,100 matches: approximately 180 seconds.[16] After this step has been completed, BNC*web* offers the option of displaying a frequency list of the sorted item(s). The top ten entries for adverb–adjective sequences are shown in Figure 4.

In addition to the total number of occurrences for each combination, Figure 4 also displays the percentage of all relevant combinations covered by each individual entry. Thus, *really good* is the most frequent combination, but its 861 occurrences only amount to less than half a per cent of all instances of (potential) adjective intensification found in the BNC. It may also be interesting – and perhaps surprising – to note that three of the top ten combinations involve the adjective *different*. Descriptive statistics of the type shown in Figures 3 and 4 often lead to serendipitous findings that open up new avenues of

[15] At the time of writing, 'sorting on node' was restricted to the first lexical item of the result string. A greater flexibility in this respect would no doubt increase the overall value of this feature (cf. Section 7).

[16] Future optimizations in the Corpus Workbench that are specifically geared towards better integration with BNC*web* may reduce this figure to a fraction of the time currently needed.

research. Apart from offering sophisticated ways of answering specific research questions, a user-friendly and feature-rich corpus tool like BNC*web* of course also lends itself well to more casual and exploratory excursions into the various facets of language use. We would therefore like to encourage the use of BNC*web* in the classroom and other contexts where innovative scientific research is not the primary motivation.



There are 78663 types and 200100 tokens in your sorted query result

No.	Lexical item(s)	No. of occurrences	Percent
1	really good	861	0.43%
2	relatively small	659	0.33%
3	slightly different	579	0.29%
4	particularly important	575	0.29%
5	completely different	512	0.26%
6	extremely difficult	488	0.24%
7	hardly surprising	446	0.22%
8	readily available	426	0.21%
9	totally different	397	0.2%
10	really nice	383	0.19%

Figure 4: *Frequency list of adverb–adjective combinations in the BNC (restricted to adverbs ending in* -ly*)*

The second sample query to be presented in this section is intended to give readers an impression of the flexibility and power of CQP syntax. At the time of writing, the BNC*web* interface has not yet been optimized to provide intuitive and user-friendly access to the full range of features offered by the Corpus Workbench. However, when "CQP syntax" is selected in the pop-up menu below the text area on the standard query page (cf. Figure 1), BNC*web* processes any well-formed CQP query entered in the search form and displays its results in the same way as described above for the simplified query. Users who have had previous experience with the Corpus Workbench as a command-line tool may thus enjoy the graphical user interface and the post-processing powers of BNC*web* without having to compromise on the complexity of their corpus searches.

The investigation of tautologies is one area of research where such an advanced query can greatly improve the precision of the query result. As typical examples, consider the italicized elements in sentences (1) and (2):

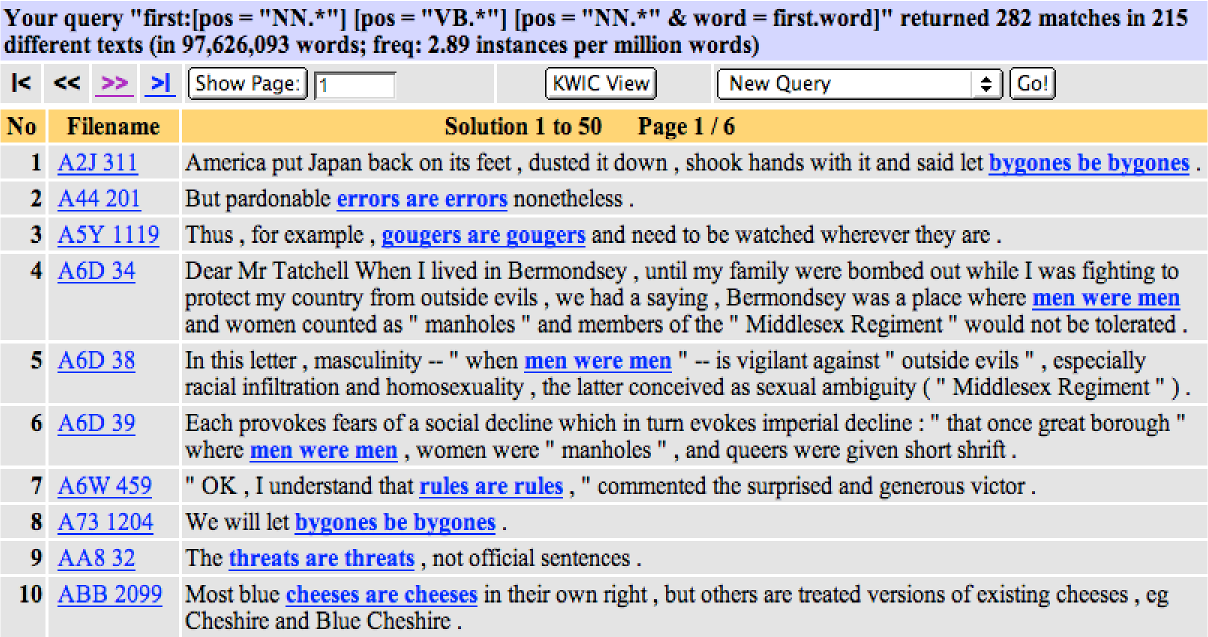
(1) It seemed out of character – but then, *facts were facts*. (H8S:4603)

(2) I mean after all *life is life*. (K21:1740)

In both cases, two instances of the same noun are linked by means of the copula *BE*. Although these sentences can of course also be retrieved by way of the simple query "N */VB* N" (i.e. 'any noun followed by any form of the verb *be* followed by any noun'), the overwhelming majority of the 19,172 resulting matches in the BNC will consist of combinations where the first and the second noun are not identical. The manual workload necessary to isolate the relevant constructions would no doubt be enormous. This situation can be avoided by making use of labels in a full-fledged CQP query, as shown below:

```
first:[pos = "NN.*"] [pos = "VB.*"] [pos = "NN.*" & word = first.word]
```

Here, the first token is required to be a noun by restricting matches to any item whose part-of-speech tag begins with "NN" (e.g. "NN1" or "NN2"). In addition, the label "first" is assigned to this token. The third token, which is again required to be a noun, is given the additional constraint that its word-form has to be identical to the word-form of the first token (which is referenced through its label "first").



Your query "first:[pos = "NN.*"] [pos = "VB.*"] [pos = "NN.*" & word = first.word]" returned 282 matches in 215 different texts (in 97,626,093 words; freq: 2.89 instances per million words)

No	Filename	Solution 1 to 50 Page 1 / 6
1	A2J 311	America put Japan back on its feet , dusted it down , shook hands with it and said let **bygones be bygones** .
2	A44 201	But pardonable **errors are errors** nonetheless .
3	A5Y 1119	Thus , for example , **gougers are gougers** and need to be watched wherever they are .
4	A6D 34	Dear Mr Tatchell When I lived in Bermondsey , until my family were bombed out while I was fighting to protect my country from outside evils , we had a saying , Bermondsey was a place where **men were men** and women counted as " manholes " and members of the " Middlesex Regiment " would not be tolerated .
5	A6D 38	In this letter , masculinity -- " when **men were men** " -- is vigilant against " outside evils " , especially racial infiltration and homosexuality , the latter conceived as sexual ambiguity (" Middlesex Regiment ") .
6	A6D 39	Each provokes fears of a social decline which in turn evokes imperial decline : " that once great borough " where **men were men** , women were " manholes " , and queers were given short shrift .
7	A6W 459	" OK , I understand that **rules are rules** , " commented the surprised and generous victor .
8	A73 1204	We will let **bygones be bygones** .
9	AA8 32	The **threats are threats** , not official sentences .
10	ABB 2099	Most blue **cheeses are cheeses** in their own right , but others are treated versions of existing cheeses , eg Cheshire and Blue Cheshire .

Figure 5: *Advanced CQP query syntax – noun–*BE*–noun constructions with identical first and second noun*

As in the case of adjective intensification, this query requires considerable time to execute (approximately 95 seconds). The result, however, is a much reduced set of only 282 matches – a mere 1.5 per cent of the more general simple query mentioned above. The first ten of these matches are displayed in Figure 5.

After having demonstrated some of the new features of BNC*web* (CQP-edition), we will now turn our attention to the future. Building on the experience gained from integrating the Corpus Workbench into BNC*web*, we will use the remaining part of this paper to present a blueprint for the development of a new and more generally applicable corpus tool.

7 The future: a white paper for C*web*

The example of BNC*web* (CQP-edition) shows that it is in fact possible to create better corpus search tools by combining the complementary strengths of existing solutions while trying to avoid their disadvantages. Heartened by our experience, we believe that further steps towards the ideal tool sketched in Section 4 can and should be made (though the ultimate goal of a perfect software can never quite be reached, of course). In this section, we outline the desired features of such a next-generation tool, which we have provisionally named CORPORA*web*, or C*web* for short. What we present here is more than a mere wish list, though. We are confident that C*web* can be realized within the foreseeable future, building on the code base of BNC*web* (CQP-edition) and our experiences with the individual tools as well as their marriage. The following is a white paper for C*web*, describing the ways in which the offspring of BNC*web* and the Corpus Workbench will be superior to its parents.

- The most severe limitation of BNC*web* (including the CQP-edition) is that it can only be used with a single corpus (viz. the British National Corpus, World Edition). C*web*, by contrast, will support a broad range of (text) corpora, provided that their structure is reasonably similar to that of the BNC.
- A crucial step in the specification of C*web* is the definition of the supported corpus annotations and formats. We envisage C*web* to be compatible with any corpus that is encoded in an XML format and whose structure conforms to the TEI (Sperberg-McQueen and Burnard 2002) and XCES guidelines.[17] In particular, the corpus text is expected to consist of

[17] The XML Corpus Encoding Standard (XCES) is an application of the TEI guidelines, which specifies explicitly how primary data and linguistic annotations should be encoded, as well as the minimum encoding level that a corpus has to achieve. The XCES specifications can be accessed on-line at <http://www.xces.org/> (27 November 2005).

"word" and "punctuation" tokens, which can be annotated with an arbitrary number of attribute-value pairs.[18] Furthermore, C*web* will expect metatextual information (i.e. metadata at text and speaker level) to be encoded in a similar fashion as in the BNC – without of course being limited to the categories represented there. In addition, arbitrary structural markup will be allowed in the form of nested XML tags (which can represent anything from document structure over layout information such as headings and lists to syntactic phrase-structure analyses).

- Even users without programming or system administration skills should be able to configure C*web* for use with a new corpus. An automated installation procedure will create the index files needed for corpus searches with CQP and also compile a number of frequency tables that are required for many of the statistical calculations carried out by the post-query features. User intervention is kept at a minimum and only relates to aspects of the system which are not vital for ensuring its basic functionality. For instance, users may choose to provide descriptions of the codes and categories used for metatextual information. In addition, they may also wish to indicate which elements of structural markup should by default be displayed in query results.
- The CQP query language appears to offer an adequate solution for most corpus searches that we can envisage at the moment.[19] However, it has to be complemented with an intuitive simplified query language for novices and casual users. One of the challenges of C*web* development is to improve the simplified query syntax of BNC*web* (CQP-edition) by extending its flexibility and generality without at the same time sacrificing ease of use. Ideally, there should be a smooth migration path from basic simplified queries over an extended syntax (hidden from the casual user) to the full-fledged CQP language.
- Both the user interface design and the range of supported post-query features will be similar to BNC*web* (viz. KWIC display, sentence display, extended context, bibliographical and/or speaker information, distribution over metatextual categories, sorting, frequency tables, collocation analysis, etc.). However, a number of optimizations are planned in order

[18] For corpora which do not exist in a TEI-compliant, tokenized format, C*web* will provide a number of user-configurable conversion routines.

[19] Again, it is impossible to support all imaginable query features in a single tool. This reservation applies in particular to any conditions involving frequency information or quantification, such as *Find a noun phrase preceded by a verb that is more frequent in the spoken part of the corpus than the written part* or *Find all nouns that are never used as an object of BUY or SELL in the corpus.* In order to answer these complex questions, a modicum of programming will always be necessary, combining existing tools into novel solutions.

to further increase its level of user-friendliness. In particular, some frequently needed basic types of analysis are currently implemented in a rather cumbersome fashion that requires the user to complete several individual processing steps. Examples are a frequency list of all matching strings (sort → by node → frequency list) or to identify adjectives collocating with a given noun (collocations → collocation settings → any adjective). While these particular features could be added as new buttons to the C*web* user interface, a more general solution is certainly desirable, in which arbitrary sequences of post-query steps can be stored as templates and accessed directly after executing a query.

- A further area of optimization concerns user management and individual customization options. A Web-based administration tool will provide a convenient way of adding new users and setting individual access restrictions that limit the amount of data that can be stored by each user (both for explicitly saved queries and for automatically cached results). This is particularly important when C*web* is installed on a central server that can be accessed by a potentially large number of concurrent users. In addition, every user will be able to change the appearance of the C*web* interface according to their personal preferences (e.g. by choosing their favourite font shape and size, as well as a suitable colour scheme).
- With respect to the overall architecture of the system, we are convinced that the client-server solution implemented for BNC*web* has the greatest merits and we therefore envisage retaining the same design for C*web*. Early versions of C*web* will again require a Unix environment for the server (such as Linux or Mac OS X, with Perl, the Apache Web server and a MySQL database), but a long-term goal is to eliminate platform-dependencies so that client and server can easily be installed on the same desktop computer from a single package. Of course, the client side will always be platform-independent, requiring no more than a modern, standards-compliant Web browser.
- Finally, C*web* should have a much more modular architecture than its parentage. Apart from the obvious benefits in terms of maintenance and development of the code base, this is particularly important in order to enable experienced and computer-savvy users (e.g. "local experts" at institutes running a C*web* server for their staff and/or students) to customize C*web* for better support of their corpora.[20] As one example, relatively simple custom XSLT stylesheets could be used to display specialized corpus annotations in a more suitable manner than the generic

[20] As has been pointed out in Section 5, the implementation of BNC*web* (CQP-edition) would have been greatly facilitated if the BNC*web* source code had had a more modular design.

built-in views. Similarly, basic programming skills in Perl would be sufficient to implement new simplified query languages that meet the specific needs of local users or are tailored to the annotations of a particular corpus.

8 Conclusion

In the introduction to this paper, we emphasized the pivotal role played by corpus tools in mediating between human researchers and their electronic objects of investigation. As our descriptions of BNC*web* and the Corpus Workbench have shown, this role can be performed in rather different ways. In their quest towards the creation of an ideal solution, the authors of such tools constantly have to negotiate the territory between a variety of opposing poles. Perhaps the most challenging of these is to meet the irreconcilable demands for a tool which is both user-friendly and intuitive to use but which at the same time imposes few limitations on the complexity and flexibility of searches. By combining the strengths of BNC*web* and the Corpus Workbench, it has been possible to create a new tool for accessing the BNC that comes at least a few steps closer to this ideal, as the two sample queries presented in Section 6 have demonstrated.

The success of our cooperation on BNC*web* (CQP-edition) has encouraged us to proceed further in the quest for the ideal corpus tool. An outline of the next stage in this development was presented in Section 7 in the form of a white paper for C*web*. Largely based on the architecture and functionality of BNC*web* (CQP-edition), this new tool will remove some of the remaining limitations (most importantly the restriction to a single corpus) and provide further improvements in user-friendliness and the handling of query results, without compromising on the versatility and efficiency of its query language.

List of References

Christ, O. (1994) "A modular and flexible architecture for an integrated corpus query system", *Papers in Computational Lexicography (COMPLEX '94)*, 22–32. Also see <http://cwb.sourceforge.net/>, accessed 6 December 2005.

Evert, S. & H. Kermes (2003) "Annotation, storage, and retrieval of mildly recursive structures", *Proceedings of the Workshop on Shallow Processing of Large Corpora (SProLaC 2003)*, ed. K. Simov & P. Osenova, Lancaster, UK: UCREL. 23-33.

Kreyer, R. & J. Mukherjee (2002) "Review of BNC*web* Version 2.0" *Linguist List* 13.2840. Available at <http://www.linguistlist.org/issues/13/13-2840.html>, accessed 20 November 2005.

Leech, G. & N. Smith (2000) *Manual to Accompany The British National Corpus (Version 2) with Improved Word-class Tagging.* Lancaster: UCREL. Available at <http://www.natcorp.ox.ac.uk/World/HTML/bnc2 postag_manual.htm>, accessed 20 November 2005.

Lehmann H.-M., P. Schneider & S. Hoffmann (2000) "BNCweb", *Corpora Galore: Analysis and Techniques in Describing English*, ed. J. Kirk, Amsterdam: Rodopi. 259-266.

Lorenz, G. (1999): *Adjective Intensification – Learners versus Native Speakers: A Corpus Study of Argumentative Writing.* Amsterdam: Rodopi.

Sperberg-McQueen, C.M. & L. Burnard (eds.) (2002): *Guidelines for Text Encoding and Interchange*. University of Oxford: Humanities Computing Unit. Also available at <http://www.tei-c.org/P4X/>, accessed 27 November 2005.