Studi Interdisciplinari su
Traduzione Lingue e Culture

Studi Interdisciplinari su Traduzione, Lingue e Culture.

Collana a cura del Dipartimento di Studi Interdisciplinari su Traduzione, Lingue e Culture (SITLeC) dell'*Alma Mater Studiorum* – Università di Bologna, sede di Forlì.

La collana del SITLeC, fondata nel 2004, raccoglie le pubblicazioni prodotte nell'ambito dell'attività scientifica dei suoi afferenti e degli studiosi che operano negli stessi ambiti a livello nazionale e internazionale. Si propone come luogo editoriale di scambio e di dialogo interlinguistico e interculturale allo scopo di diffondere e rendere disponibili, a livello cartaceo e/o su supporto elettronico, i risultati della ricerca in diverse aree, come la linguistica teorica e applicata, la linguistica dei *corpora*, la terminologia, l'interpretazione, gli studi di genere, la critica letteraria, il teatro, gli studi culturali, gli studi sull'umorismo, privilegiando le dimensioni del tradurre, inteso come luogo di incontro e di scontro fra lingue e culture.

Tutte le pubblicazioni sono approvate dal Consiglio di Dipartimento, sentito il motivato parere di almeno due esperti qualificati esterni.

Il responsabile della Collana è il Direttore del SITLeC, affiancato da un comitato scientifico internazionale, articolato e variabile in relazione alle tematiche trattate.

WaCky!
Working Papers on the Web as Corpus

edited by
Marco Baroni and Silvia Bernardini

Gedit Edizioni

Collana Studi Interdisciplinari su Traduzione, Lingue e Culture
Direttore: Michele Prandi

Volume pubblicato con il contributo del Dipartimento di Studi Interdisciplinari su Traduzione Lingue e Culture (SITLeC) dell'*Alma Mater Studiorum* – Università di Bologna, sede di Forlì, Corso Diaz, 64 – 47100 Forlì.

ISBN 88-6027-004-9

First edition: September 2006

Gedit Edizioni
Via Irnerio 12/5
40126 Bologna
tel 051 4218740 fax 051 4210565
copertina: Avenida, Modena
stampa: Editografica, Rastignano (BO)

The volume was typeset in LaTeX by the editors and authors.

This collection of working papers puts together presentations at two Web as Corpus workshops (Forlì, January 14, 2005, Birmingham, July 13, 2005), and articles that were born out of discussions and collaborative experimentation among the WaCky community members. WaCky (for "**W**eb **a**s **C**orpus **k**ool **y**nitiative", in case you were wondering...) is a project started informally (i.e., with very little funding...) in 2003. It brings together linguists who think the World Wide Web is a great resource for their research, and that it would be even greater if it could be annotated and interrogated in a more linguist-friendly way. While we are aware that the task is an awesome one, we also believe that it is one worth putting some of our time and efforts into, and that interim results (e.g., the billion word, annotated, Web-derived corpora that have already seen the light for German and Italian) may equally provide very rich resources to study languages (on and off the Web). Through the publication of this collection of papers we hope to raise the interest of other researchers worldwide, who wish to contribute to this challenge.

For more information on WaCky or to participate in the initiative, please visit the WaCky wiki: `http://wacky.sslmit.unibo.it/`

We gratefully acknowledge the Fondazione Cassa dei Risparmi di Forlì for financial help in organizing the first Web as Corpus workshop. We also would like to thank the participants in the Web as Corpus workshops and in the online WaCky community – in particular, the contributors to this volume and Adam Kilgarriff – for very stimulating discussions.

Marco Baroni
Silvia Bernardini

List of Contributors

Marco Baroni SSLMIT, Università di Bologna, Corso della Repubblica 136, Forlì, 47100 Italy; baroni@sslmit.unibo.it

Silvia Bernardini SSLMIT, Università di Bologna, Corso della Repubblica 136, Forlì, 47100 Italy; silvia@sslmit.unibo.it

Sara Castagnoli SITLeC, Università di Bologna, Corso Diaz 64, Forlì, 47100 Italy; scastagnoli@sslmit.unibo.it

Massimiliano Ciaramita Yahoo! Research Barcelona, Ocata 1, 1st floor, 08003 Barcelona, Catalunya, Spain; massi@yahoo-inc.com

Thomas Emerson Gerson Lehrman Group, 2 Oliver Street, 7th Floor, Boston, MA 02109, USA; tree@glgroup.com

Stefan Evert Cognitive Science Institute, University of Osnabrück, Albrechtstr. 28, 49069 Osnabrück, Germany; stefan.evert@uos.de

Claudio Fantinuoli EURAC Research, Viale Druso 1, Bolzano, 39100 Italy; claudio.f@gmx.de

Rüdiger Gleim Bielefeld University, D-33615 Bielefeld, Germany; ruediger.gleim@uni-bielefeld.de

Alexander Mehler Bielefeld University, D-33615 Bielefeld, Germany; alexander.mehler@uni-bielefeld.de

John O'Neil Basis Technology, Inc., 150 CambridgePark Drive, Cambridge, MA 02140, USA; oneil@basistech.com

Serge Sharoff Centre for Translation Studies, School of Modern Languages and Cultures, University of Leeds, Leeds, LS2 9JT, UK; s.sharoff@leeds.ac.uk

Motoko Ueyama SSLMIT, Università di Bologna, Corso della Repubblica 136, Forlì, 47100 Italy; motoko@sslmit.unibo.it

Contents

A WaCky Introduction

Silvia Bernardini, Marco Baroni and Stefan Evert

1 The corpus and the Web

We use the Web today for a myriad purposes, from buying a plane ticket to browsing an ancient manuscript, from looking up a recipe to watching a TV program. And more. Besides these "proper" uses, there are also less obvious, more indirect ways of exploiting the potential of the Web. For language researchers, the Web is also an enormous collection of (mainly) textual materials which make it possible, for the first time ever, to study innumerable instances of language performance, produced by different individuals in a variety of settings for a host of purposes.

One of the tenets of corpus linguistics is the requirement to observe language as it is produced in authentic settings, for authentic purposes, by speakers and writers whose aim is not to display their language competence, but rather to achieve some objective through language. To study "purposeful language behavior", corpus linguists require collections of authentic texts (spoken and/or written). It is therefore not surprising that many (corpus) linguists have recently turned to the World Wide Web as the richest and most easily accessible source of language material available. At the same time, for language technologists, who have been arguing for long that "more data is better data", the WWW is a virtually unlimited source of "more data". The potential uses to which the Web has been (or can be) put within the field of language studies are numerous and varied, from checking word frequencies using Google counts to constructing general or specialized corpora

of Web-published texts. The expression "Web as corpus" is nowadays often used to refer to these different ways of exploiting the WWW for language studies.

In what follows we briefly consider four different ways of using the Web as a corpus, focusing particularly on those taking the lion share of this volume of working papers: the Web as a "corpus shop", and the "mega-corpus/mini-Web" as a new object. The latter in particular will be described in some detail, and special attention will be paid to the design of this resource and the challenges posed by its development.

2 *Web as Corpus (WaC)*: four senses

There is currently no unified understanding of the expression Web as corpus. We have identified four separate senses, though there are probably others:

1. The *Web as a corpus surrogate*
2. The *Web as a corpus shop*
3. The *Web as corpus proper*
4. The *mega-corpus/mini-Web*

Researchers (and users in general) using the Web as a corpus *surrogate* turn to it via a standard commercial search engine for opportunistic reasons. They would probably use a corpus, possibly through corpus analysis software, but none exists for their purposes (e.g., because available corpora are too small), or they do not have access to one, or they do not know what a corpus is. The translator trainees at the School for Interpreters and Translators, University of Bologna (Italy), for instance, use the Web as a reference tool in their translation tasks, though the search

is often time consuming, the relevance and authoritativeness of the solutions found is hard to assess, and the observation of recurrent patterns very difficult. It would make sense for them to use a corpus, if one existed as large as the Web, and if they knew how to use it.[1] Similarly, researchers who rely on Google-like hit counts for their studies (e.g., Chklovski and Pantel 2004) live with the brittleness[2] and reduced flexibility of the search engine, though they would no doubt prefer a more stable resource, allowing replication and providing facilities for more sophisticated queries. Linguist-oriented meta-search engines like KWiCFinder[3] and WebCorp[4] wrap around the standard output of Web search engines some of the features and facilities of corpus search engines (e.g., the KWIC format, a collocation tool, and so forth). Though this solution leaves questions linked to the datasets and retrieval strategies untouched, users can to some extent pretend to be consulting the Web in a corpus-like environment.

Others using the Web as a corpus treat it as a corpus *shop*. They query a traditional search engine for combinations of search words, taking advantage of the facilities offered by the engine (e.g., selection of language, provenance, URL-type etc.) to focus their queries. They (select and) download the texts retrieved by the engine, thus creating a corpus in the traditional sense of the term. This procedure, which can be automatized to various degrees, is adopted by those who require specialized corpora, e.g., for translation, terminology or text analysis purposes. Several researchers have discussed the didactic advantages of "disposable" corpora (e.g., Varantola 2003) in the teaching of foreign languages and translation skills. Castagnoli (this volume) describes a classroom

[1] Unless stated otherwise, by "Web" and "corpus" we refer to both the text materials and the search engines used to index and search them.

[2] `http://aixtal.blogspot.com/2005/02/web-googles-missing-pages-mystery.html`

[3] `http://miniappolis.com/KWiCFinder/KWiCFinderHome.html`

[4] `http://www.webcorp.org.uk/`

experience where learners a) use the BootCaT toolkit (Baroni and Bernardini 2004), a set of Unix tools, to construct corpora from specialized domains in a semi-automated way, b) evaluate the corpora and c) use them to investigate terminology and retrieve typical instances of usage in context. Castagnoli suggests that the limits of this automatic procedure can be turned into advantages in a pedagogic context, where learners can be made to reflect on their text selection strategies and documentation skills. The development of a Web interface for the BootCaT tools (Baroni et al. 2006) should remove the technical hurdles for less computer-literate users and favor a more widespread use in the classroom and among language professionals.

Fantinuoli (this volume), Sharoff (this volume) and Ueyama (this volume) also use the BootCaT tools, but take a more descriptively oriented perspective. Their aim is an evaluation of corpora constructed semi-automatically from the Web. While Fantinuoli (like Castagnoli) focuses on the construction of specialized corpora for language professionals, Sharoff uses this methodology to build general language corpora of Chinese, English, German and Russian, and Ueyama of Japanese. These authors focus on different ways of evaluating their products: the comparison between manually and automatically constructed corpora and manually and automatically extracted terms in the case of Fantinuoli, the qualitative and quantitative observation of topics, genres and lexical items in Web corpora built in different ways in Ueyama, and the comparison of word frequency lists derived from the Web and from traditional corpora in Sharoff. These articles contribute to the establishment of good practices and open the way to the empirical study of a set of still under-investigated questions such as: In what ways can we say that traditional corpora differ from Web-derived corpora? How does the corpus construction methodology affect the nature of the resulting corpus?

So far we have been concerned with ways of using the Web opportunistically, to derive generalizations about (subsets of) a language either directly through search engine queries or indirectly through the downloading of Web published texts. For these purposes paper texts would equally be appropriate, if not for the obstacle of digitizing them. Our third possible meaning of the notion of *Web as corpus*, the "Web as corpus proper", is different inasmuch as it purports to investigate the nature of the Web. In the same way as the British National Corpus aims to represent the whole of British English at a given point in time, it is possible to envisage a corpus that represents Web English at a given point in time.[5] This research paradigm could tell us something about the language used on the Web (glimpses of this are provided in this volume by Ueyama and Sharoff). Clearly, extensive discussion and experimentation is needed to develop criteria for Web sampling. Input might come from taxonomy-oriented surveys (along the lines of the article by Mehler and Gleim in this volume). We expect this area of research to feature prominently in WaC linguistics in the next few years.

Lastly, our fourth and most radical way of understanding the expression *Web as a corpus* refers to attempts to create a new object, a sort of mini-Web (or mega-corpus) adapted to language research. This object would possess both Web-derived and corpus-derived features. Like the Web, it would be very large, (relatively) up-to-date, it would contain text material from crawled Web sites

[5]One could argue that this sense of Web as corpus is somewhat different from those discussed so far (and indeed from the one discussed below). After all, the *corpus surrogate* and the *corpus shop* approaches are different ways of using Web data for similar purposes (to investigate linguistic issues), whereas in the WaC proper approach the purposes of the investigation differ (i.e., we are trying to learn about the Web, rather than using the Web to learn about language). We include this sense here anyway because the aim is simply telling different understandings of the expression apart, rather than providing a consistent classification.

and it would provide a fast Web-based interface to access the data. Like a corpus, it would be annotated (e.g., with POS and lemma information), it would allow sophisticated queries, and would be (relatively) stable. Both people wanting to investigate aspects of language through the Web, and people wanting to investigate aspects of the Web through language could profit from this corpus.

We are convinced that this is a valuable research project because it answers a widely-felt need in our community of (computational) linguists, language and translation teachers and language professionals for a resource that combines the reliability and the flexibility of corpora and their search tools with the size, variety and timeliness of the Web. The chances that commercial Web search engines be interested in such a research agenda are very low, and relying on the less standard facilities they offer may not be a good idea in the long run.[6]

Besides being valuable, we believe that this is also a feasible, though challenging, endeavor. The present authors and several contributors to this volume are currently involved in the piloting of very large Web-derived corpora in English, German and Italian, in a project (started at the end of 2004) that emphasizes the development and sharing of open tools and resources. A series of workshops have been organized which have provided a public discussion space (the Web as corpus workshop in Forlì, January 14, 2005; the Web as corpus workshop at CL05 in Birmingham, July 14, 2005; and the Web as corpus workshop at EACL, Trento, April 3, 2006). Discussion is constantly taking place also through the project wiki, the so-called *WaCky wiki*.[7] Many WaCky contributors are actively involved in the recently established Special Interest Group on Web as Corpus (SIGWAC) of the Association for

[6] AltaVista discontinued the NEAR operator in 2004. The Google API keys (on which the BootCaT tools currently rely) have provided very discontinuous functionality during tests carried out in the last few months of 2005 and in early 2006.

[7] `http://wacky.sslmit.unibo.it/`

Computational Linguistics (ACL).[8] At the same time, infrastructure building has also started in Forlì, with the aim to secure the minimum technical prerequisites to begin the piloting phase. Two mega corpora are at an advanced stage of development (*deWaC* and *itWaC*, for the German and Italian languages, respectively: Baroni and Kilgarriff 2006; Baroni and Ueyama 2006), the construction of other corpora is under way for other languages (English, Chinese and Russian), and more funds to proceed with the project are being sought.

Among the papers in this collection, the one by Emerson and O'Neil presents in detail the first steps of data collection for the purposes of building a mega corpus of Chinese. There are two main sides that are relevant to the construction of a mega corpus/mini-Web. First, one has to retrieve, process and annotate Web data. Second, one has to index these data, and construct an interface to allow prospective users to access the data. In the next two sections, we will present our ideas about both aspects of the process, relying on our experiences with deWaC and itWaC and on the work reported in the remainder of this volume.

3 Constructing Web corpora

The basic steps to construct a Web corpus are:

1. Select the "seed" URLs
2. Retrieve pages by crawling
3. Clean up the data
4. Annotate the data

We discuss each of these in turn.

[8] `http://www.sigwac.org.uk/`

3.1 Selecting seed URLs

The crawl has to start from a set of seed URLs. For special-purpose corpora, it is relatively straightforward to decide the seeds (e.g., if one wants to build a corpus of blogs, one can select a random set of blog URLs from one or more blog servers). For a general-purpose corpus, one would ideally want to draw a random sample of pages that are representative of the target language. As discussed in the article by Ciaramita and Baroni, this is not the same as drawing a random sample of webpages. For example, suppose that the Italian Web is composed of a 90% of pornographic pages, 9% of Linux *howtos*, and that all other text types together make up just 1% of the whole. For the purpose of building a corpus, we would probably prefer a sampling method heavily biased in favor of selecting from this 1%, rather than a true random sample that would lead to a corpus of mostly pornography plus the occasional bash shell guide. The fact that the notion of "representativeness" of a corpus (and how to measure it) is far from well-understood (Kilgarriff and Grefenstette 2003) complicates matters further. Ciaramita and Baroni propose a measure of "unbiasedness" of a Web-derived corpus based on the comparison of the word frequency distribution of the target corpus to those of deliberately biased corpora.

Both Sharoff and Ueyama select seed URLs by issuing (automated) queries for random content word combinations to Google, and retrieving the URL lists returned by the engine. The qualitative evaluation carried out by Sharoff suggests that the variety (in terms of parameters such as genre and domain) of the collections of documents corresponding to these URLs is closer to what we would expect from a balanced corpus than to what we find in biased collections, such as newswire corpora. An important aspect of this methodology is how the words used in the queries are selected. Ueyama's experiments suggest that selecting words from traditional corpora might bias the queries towards pages containing

higher register prose and "public life" domains, thus missing some of the most interesting linguistic material available on the Web – non-professionally written, informal prose on everyday topics. Pages of this sort can be found using words from basic vocabulary lists. The seed URLs chosen to build the WaCky initiative German and Italian corpora were retrieved from Google with combinations of words extracted both from traditional newspaper corpora and from "basic vocabulary" lists for language learners, in the hope to tap into both higher register/public and lower register/private sections of the Web.

Emerson and O'Neil select URLs matching their target language (Chinese) from the Open Directory Project (ODP),[9] a large, open directory of webpages maintained by volunteers. This method has several advantages over the former: It does not rely on a commercial enterprise such as Google, and the metadata information provided by ODP can be exploited for sampling. On the negative side, the set of URLs listed by ODP is much smaller than the set indexed by Google (at the moment of writing, about 5 million vs. 8 billion). Moreover, ODP seems biased in favor of top level pages, whereas the pages retrieved by random content word queries to Google often come from deeper layers of websites, and as such tend to be characterized by richer textual content. Devising and comparing seed URL selection strategies will be an important area for future WaC research.

3.2 Crawling

If the list of seed URLs is long and/or one does not aim to build a very large corpus, crawling can be as simple as retrieving the documents corresponding to the seed URLs (this is what Sharoff and Ueyama do). Otherwise, one uses the seed URLs to start a crawl of the Web, i.e., a program is launched that retrieves pages

[9] `http://www.dmoz.org`

corresponding to the seed URLs, extracts new URLs from the links in the retrieved pages, follows the new links to retrieve more pages, and so on. Conceptually, crawling is a straightforward procedure; however, only a sophisticated implementation of the procedure will allow one to perform a successful large-scale crawl. There are several issues that need to be addressed.

- *Efficiency*: As more pages are retrieved, the queue of discovered URLs grows very large. Thus, the crawler must be able to manage such a large list in a memory-efficient way.
- *Duplicates*: The crawler must make sure that only URLs that have not been seen already are added to the list.
- *Politeness*: Minimally, the crawler must respect the directives specified by webmasters in a site's `robots.txt` file. However, it should also avoid hammering the same site with thousands of requests in a short time span, and provide an easy way to contact the owner of the crawl.
- *Traps*: The crawler should avoid "spider traps", i.e., malicious sites that try to stop it, e.g., by luring it into a loop in which it will continue downloading dynamically generated pages with random text forever (not a good thing for corpus building!)
- *Customization*: The crawler should be easy to customize (e.g., for a linguistic crawl one might want to limit the crawl to pages from a certain national domain, and focus on HTML documents) and, given that a large crawl will probably take weeks to complete, it should be possible to monitor an ongoing crawl, possibly changing parameters on the fly.
- *File handling*: Finally, given that a large crawl will retrieve millions of documents, the crawler should handle the retrieved data in an intelligent manner (on the one hand, we

would not want to have to deal with millions of output files; on the other, a single file of a few hundreds gigabytes would also be hard to manage).

For all these reasons, simple tools such as the Unix utility `wget` are not appropriate for large-scale crawls, and programs specifically designed for such task should be used. The crawl described in Emerson and O'Neil's article is based on one such tools, i.e., Heritrix, the open-source Java crawler developed at the Internet Archive.[10] Heritrix is also employed by the WaCky project.[11]

3.3 Data cleaning

Once the crawl is over, we are left with a (possibly very large) set of HTML documents,[12] and we have to convert them into something that can be used as a linguistic corpus. For many purposes, HTML code and other non-linguistic material should be removed. Presumably, language/encoding detection and (near-)duplicate discarding are desirable steps independently of the purposes of the corpus.

An interesting side effect of WaC activities is that, because Web data are so noisy, data cleaning must take center stage, unlike in traditional NLP, where it has been seen as a minor preprocessing step that is not really worth talking about (standard introductions to NLP, such as Jurafsky and Martin 2000 and Manning and Schütze 1999, do not dedicate any space to the topic). In-

[10] `http://crawler.archive.org`

[11] An alternative fully featured crawler is the Ubicrawler (`http://ubi.imc.pi.cnr.it/projects/ubicrawler`), which, however, at the moment of writing does not appear to be publicly distributed under a GNU-like license, and, consequently, is not supported by the same kind of wide community that supports Heritrix.

[12] Other formats, such as Acrobat's PDF and Microsoft's doc might also be converted to text and added to the corpus. We do not discuss this possibility here.

deed, the Special Interest Group on Web as Corpus of ACL is currently preparing a competitive data cleaning task, CLEANEVAL, as its first public activity.[13]

3.3.1 HTML code removal and boilerplate stripping

Tools such as `vilistextum`[14] (used by Emerson and O'Neil) and the standard Unix textual browser `lynx` (used by Sharoff) extract plain text from an HTML document, while attempting to preserve its logical structure and the hyperlink information. This is appropriate for certain purposes – e.g., to parse a document according to the "graph grammar" of webpages proposed in Mehler and Gleim's chapter (indeed, for such purposes it might be desirable to preserve the HTML code itself). Logical structure and hyperlink information might also be useful for purposes of document categorization. However, structural markup and links will constitute noise for the purposes of further linguistic processing (tokenization, POS tagging, etc.).

Equally problematic, in terms of linguistic processing and extraction of linguistic information, is the presence of "boilerplate", i.e., the linguistically uninteresting material repeated across the pages of a site and typically machine-generated, such as navigation information, copyright notices, advertisement, etc. Boilerplate can clutter KWIC displays, distort statistics and linguistic generalizations (we probably do not want "click here" to come up as the most frequent bigram of English), and make duplicate detection harder. Boilerplate is harder to identify than HTML/javascript, since it is regular text, not overtly delimited code. For corpora based on crawls of a limited number of domains, it might be possible to analyze pages from the domains and manually develop regular expressions to spot and remove boilerplate. For larger crawls,

[13] `http://cleaneval.sigwac.org.uk/`
[14] `http://bhaak.dyndns.org/vilistextum`

domain-independent methods must be applied. For the development of the WaCky corpora, we used HTML tag density as a fast heuristic method to filter out boilerplate (re-implementing the algorithm of the Hyppia project BTE tool).[15] The idea is that the content-rich section of a page will have a low HTML tag density, whereas boilerplate text tends to be accompanied by a wealth of HTML (because of special formatting, many newlines, many links, etc.) Thus, of all possible spans of text in a document, we pick the one for which the quantity $\text{Count}(tokens) - \text{Count}(tags)$ takes the highest value.

If we are interested in the Web as a source of linguistic samples, boilerplate stripping is fundamental. If we are studying the make-up of HTML documents and their linking structures, boilerplate stripping might be undesirable, as it might destroy the logical structure of a document. Optimally, a Web-based corpus should satisfy both needs by providing access to the original, unprocessed HTML documents as well as to a linguistically annotated version that had code and boilerplate removed.

3.3.2 Language/encoding detection

For Western European languages, language filtering can be a simple matter of discarding documents that do not contain enough words from a short list of function words (this is the strategy we employed when building the German and Italian WaCky corpora). For other languages, encoding detection must be performed together with language filtering, since webpages in the same languages can come in a number of different encodings. Free tools such as the TextCat utility[16] or the proprietary tools used by Emerson and O'Neil can perform this task. Tools such as the `recode` utility[17] can then convert all pages to the same encod-

[15] `http://www.smi.ucd.ie/hyppia/`
[16] `http://odur.let.rug.nl/~vannoord/TextCat`
[17] `http://recode.progiciels-bpi.ca/`

ing for further processing. Language detection will typically work poorly if the HTML code has not been stripped off. Moreover, if the detection algorithm relies on statistical models extracted from training data (often, character n-grams), these training data should not be too dissimilar from the Web data to be analyzed. For example, when using TextCat on German Web data, we noticed that the tool systematically failed to recognize German pages in which nouns are not capitalized – an informal way of spelling that is common on the Web, but virtually unattested in more standard sources such as newspaper text. A more difficult issue is that of dealing with pages that contain more than one language – however, given the wealth of data available from the Web, it might be sufficient to simply discard such pages (assuming they can be identified). Lastly, word lists can be used to identify and discard "bad" documents in the target language (e.g., pornography and Web-spam).

3.3.3 (Near-)duplicate detection

Identical pages are easy to identify with fingerprinting techniques. However, crawl data will typically also contain *near duplicates*, i.e., documents that differ only in trivial details, such as a date or a header (e.g., the same tutorial posted on two sites with different site-specific material at the beginning and/or end of the document). In principle, near duplicates can be spotted by extracting all n-grams of a certain length (e.g., 5-grams) from each document and looking for documents that share a conspicuous amount of such n-grams. However, for large corpora a procedure of this sort will be extremely memory- and time-consuming. Standard methods have been developed within the WWW-IR community (see, e.g., Broder et al. 1997 and Chakrabarti 2002) to obtain an estimate of the overlap between documents on the basis of random selections of n-grams in a very efficient manner. These techniques can also be used to find near duplicates in linguistic Web cor-

pora (a simplified version of Broder's method has been used in the clean-up of the WaCky corpora).

Notice that near duplicate spotting will work better if boilerplate stripping has been performed, as boilerplate is a source of false positives (documents that look like near duplicates because they share a lot of boilerplate) as well as false negatives (near duplicates that do not look as similar as they should because they contain different boilerplate). A more delicate issue concerns document-internal duplicate detection, e.g., pages downloaded from a bulletin board that contain a question and several follow-up postings with the question pasted into the replies. Not only can this sort of duplication be hard to spot, but its removal might disrupt the discourse flow of the document. Indeed, one might wonder whether removal of document-internal duplication is a theoretically justified move.[18]

3.4 Annotation

Tokenization, POS annotation and lemmatization of a Web corpus that has undergone thorough clean-up are straightforward operations. However, one has to be careful about the peculiarities of Web language, such as smileys, non-standard spelling, high density of neologisms, acronyms, etc. Ideally, tokenizing rules should take these aspects into account, and POS taggers should be re-trained on Web data.

The diversified, ramshackle nature of a crawled Web corpus means metadata are at the same time sorely needed (who is the author of a page? is the author a native speaker? what is the page about?) and difficult to add, both for technical reasons (the sheer size of the corpus) and because the Web presents a wealth of new "genres" and peculiar domains (how do you classify a page

[18]Indeed, if the corpus is seen as a random sample of the Web, any form of (near-)duplicate removal becomes a theoretically dubious move.

about 9/11 written by a religious fanatic that is half blog and half advertisement for his book?)

The articles of Sharoff and Ueyama in this volume report manual classification of samples of Web corpus documents in terms of genre, domain and other parameters. It is clear from these preliminary investigations that the categories used to classify traditional corpora, often based on library classification systems, have to be extended and revised in order to account for Web data. At the very least, the sizable proportion of "personal life" domains and genres present on the Web requires a fine grained taxonomy that is not present in traditional corpora (since they typically do not contain many specimens of this sort). In order to annotate the whole corpus, rather than a sample, one has of course to use automated, machine-learning techniques and, for very large corpora, efficient methods will have to be adopted (see, e.g., Chakrabarti et al. 2002).

While Sharoff and Ueyama categorize their Web corpus on a document-by-document basis, as one would do with a traditional corpus, Mehler and Gleim propose a rich representational system for Web hypertext, acknowledging that, to find meaningful textual units on the Web, we must look at whole webpages, which may or may not be spread across multiple HTML files. Again, we see here a difference in purpose. Traditional document-level annotation is probably more appropriate if we see the Web as a very rich source of data for what is ultimately to be used as a traditional corpus representative of a specific natural language, whereas Mehler and Gleim's approach looks at Web text as an object of study in itself. In any case, to annotate a connected set of Web documents according to Mehler and Gleim's proposal, automated categorizations techniques are also needed. Whether and how the complex, layered structures proposed by these authors can be induced using machine-learning techniques is an interesting topic for further research.

4 Indexing and searching a Web corpus

After the collection and linguistic annotation of a Web corpus as detailed in section 3, the data will typically be available as a collection of plain text files, usually in one-word-per-line format or in some XML format. The rich amount of authentic linguistic evidence provided by such a corpus, however, is useful only to the extent that the phenomena of interest can be retrieved from the corpus by its users. Like data cleaning, tools to store and retrieve linguistic data have been somewhat overlooked in traditional NLP work. However, they become fundamental when handling very large Web-derived corpora, where the classic ad hoc "disposable retrieval script" approach often adopted by computational linguists does no longer look like an attractive option. Development of indexing and retrieval software featuring a powerful query syntax and a user-friendly interface is probably the area in which most work still needs to be done, before we can start seriously thinking of a fully fledged "linguist search engine". Indeed, articles in this collection deal with nearly all other aspects of WaC work, but this is an area that is virtually unexplored by our authors. Consequently, we dedicate the longest section of this introduction to this topic.

In general, corpus data can be exploited in two ways: either by sequential processing (a typical example would be unsupervised training of a statistical NLP tool or a linguist reading through a corpus sentence by sentence), or by targeted search for a certain linguistic phenomenon (typically a particular lexical and/or syntactic construction). This type of search is often called a *corpus query*. A second distinction can be made between *online* processing, which is fast enough to allow interactive refinement of searches (for this purpose, query results should be available within a few seconds, or at most several minutes) and *offline* processing, where a task is started by the user and results might be ready for inspec-

tion after several hours or even days. For most linguistic purposes, the focus will be on online corpus queries.

In the following subsections, we discuss the requirements for an online query tool for Web corpora, henceforth called the *WaCky query engine*. Section 4.1 introduces four general requirements on software for the linguistic exploitation of Web corpora, as a basis for the ensuing discussion. Section 4.2 addresses the expressiveness of the query language itself, followed by the related technical issues of corpus storage and indexing strategies in section 4.3. Finally, section 4.4 argues for the combination of corpus queries with precompiled frequency databases, drawing on the advantages of both online and offline processing. The diverse components of the WaCky query engine can then be integrated seamlessly under a uniform Web interface that provides a familiar and convenient front-end for novice and expert users alike.

4.1 Requirements for linguistic search

The main challenge that online query tools for Web corpora face is to find a good balance between several conflicting requirements:

1. *Expressiveness*: The query tool should offer a flexible query language that allows the formulation of sophisticated queries to identify complex linguistic patterns in the corpus.

2. *Ease of use*: It should present a convenient and intuitive front-end to novice users.

3. *Performance*: It should support fast online searches, with response times that are short enough for interactive work even on very large corpora.

4. *Scalability*: It should be able to handle Web corpora of a billion words and more.

Different query tools will satisfy these requirements to varying degrees, focusing either on expressiveness or on speed and convenience. The two extremes of the range of possible approaches are perhaps best embodied by the Google search engine on the one hand (focusing on requirements 2–4) and the Nite XML Toolkit[19] on the other (focusing on requirement 1). Google searches several hundred billion words with ease and often returns the first page of matches within less than one second. However, it is restricted to simple Boolean queries on word forms, i.e., queries which test for the co-occurrence or non-co-occurrence of given word forms within the same document (a webpage, PDF file, etc.).[20] In contrast to this, the query language of the Nite XML Toolkit allows for complex logical expressions that build on multiple layers of equally complex linguistic annotations, but the current implementation is only able to handle corpus sizes well below 100,000 words.

The discussion in the following subsections depends to some extent on the type and complexity of annotations that have to be supported. Hence we will briefly sketch our assumptions about the annotations of Web corpora. Following section 3.4, we understand a Web corpus essentially as a sequence of word form tokens annotated with linguistic interpretations such as POS tags and lemmas. As pointed out there, meta-information about the speaker/writer of a text, its language, date of publication, genre, etc. is crucial for many applications. In most cases, such metadata can be represented as simple attribute-value pairs attached to the documents in the corpus (or to individual paragraphs, e.g., when a document contains text in different languages). In addition to this most basic

[19] See section 4.2 below

[20] For some languages, including English, Google also performs stemming, i.e., it attempts to remove morphological suffixes from search terms to broaden the search. However, since stemming is not performed in a linguistically consistent way and since it is not clear whether stemming can be disabled/enabled explicitly (i.e., to search for literal word forms), this renders the Google search engine unsuitable for many kinds of linguistic research.

set of linguistic annotations, shallow structural markup – ranging from text structure (paragraphs, sentences, lists, tables, etc.) to non-recursive chunk parsing – can significantly facilitate corpus queries and can be added by automatic methods with sufficient accuracy and efficiency. We will therefore also assume that many Web corpora contain such structural markup, with start and end points of structures indicated by non-recursive XML tags in the text or XML files.

Many users would certainly also like to have access to complete syntactic analyses of all sentences in the corpus, in the form of parse trees or dependency graphs. Such highly structured datasets put greater demands on the internal representation of corpus data, and require a fundamentally different type of query language than token-based annotations. Currently, we are not aware of any automatic tools that would be able to perform deep syntactic analysis with the required accuracy and coverage,[21] and the computational complexity of state-of-the-art systems leads to parse times ranging from several seconds to several minutes per sentence, rendering them unsuitable for the annotation of billion-word Web corpora.[22] Therefore, in the following discussion we assume that Web corpora are not annotated with complex syntactic structures. While this assumption reduces the demands on representation formats, it also means that the query language will have to provide sophisticated search patterns to make up for the lack of pre-annotated syntactic information.

[21] A recent statistical parser for German (Schiehlen, 2004) achieves F-scores between 70% (for constituents) and 75% (for dependency relations). While this level of accuracy might be sufficient for information retrieval and training of statistical NLP models, it does not provide a reliable basis for linguistic corpus queries.

[22] A syntactic parser that manages to analyze on average one word per second (which is faster than most systems that we have tested on current off-the-shelf hardware), would take 30 years to annotate a billion-word corpus.

4.2 A query tool for Web corpora

There is a wide range of query languages and implementations, which can be used for linguistic searches of different complexity. Here, we summarize the four most common approaches to corpus queries and discuss their suitability for the WaCky query engine.

The simplest method is *Boolean search*, which identifies documents that contain certain combinations of search terms (expressed with the Boolean operators AND, OR and NOT, hence the name). This basic type of Boolean search is exemplified by the Google search engine. More advanced implementations such as that provided by the open-source search engine Lucene[23] allow wildcard patterns for individual terms, constraints on metadata and to some extent also on linguistic annotations, and proximity searches for terms that occur near each other (similar to AltaVista's famous but discontinued NEAR operator). From the perspective of Web corpora, such tools can be used to build a simple concordancer that looks up individual keywords or phrases with optional metadata constraints. Proximity search allows for some variation in the phrases, and, with access to linguistic annotations, generalizations can also be expressed (e.g., that one of the words in a phrase is an arbitrary noun rather than a particular one). However, the Boolean combination of search terms is primarily designed to find documents about a particular topic (for information retrieval purposes) and will rarely be useful to linguists (although it could be used to identify simple collocational patterns).

Most of the currently available query engines for large corpora build on a *regular query language*.[24] Prominent implementations are the IMS Corpus WorkBench (CWB) with its query processor

[23] `http://lucene.apache.org/`

[24] "Regular" is used here as a the technical term from formal language theory, i.e., referring to patterns that can be described by regular expressions and finite-state automata.

CQP, the similar Manatee corpus manager (which is now part of the Sketch Engine) and Xaira, the successor of the SARA query tool supplied with the British National Corpus.[25] All three implementations are available under the GPL license. Regular query languages typically use regular expressions at the level of words and annotation values, and similar regular patterns to describe contiguous sequences of words (CQP and Manatee use a basic regular expression syntax for these patterns, but queries could also take the form of non-recursive rewrite-rule grammars, e.g., through use of CQP's built-in macro language). Many of these query languages extend the basic regular patterns. They may provide support for shallow structural markup, e.g., by inserting XML start and end tags in the query expressions. In CQP, matching pairs of start and end tags can be used to express shallow nesting of structures (e.g., PP within NP within S). Query languages will often also allow constraints on metadata, either appended to a query expression as "global constraints" or by pre-selecting a subcorpus of suitable documents for the search.

Some systems go one step further and allow queries to be formulated as *context-free grammars*. Unlike regular languages, this approach can identify recursively nested patterns of arbitrary complexity.[26] In addition, linguists often find it more intuitive to describe a search pattern with familiar context-free phrase-structure rules than to formulate an equivalent regular expression pattern (even when recursion is not required). Gsearch[27] is an *offline* corpus query tool based on context-free grammars, which is also

[25] More information about these tools can be found at the following URLs: `http://cwb.sourceforge.net/` (CWB), `http://www.textforge.cz/download.html` (Manatee), `http://www.sketchengine.co.uk/` (Sketch Engine), and `http://xaira.sourceforge.net/` (Xaira).

[26] As an example for such a structure, consider German noun phrase chunks, which may – at least in principle – contain an unlimited number of recursively nested, center-embedded noun phrases.

[27] `http://www.hcrc.ed.ac.uk/gsearch/`

available under the GPL license. We are currently not aware of any software using context-free rules for online queries.

In order to make use of deep linguistic analyses such as parse trees or dependency structures, *graph-based query languages* interpret a corpus together with all its annotations as a directed acyclic graph. Implementations of a graph-based query language include `tgrep`,[28] and the more recent TIGERSearch[29] and Nite XML Toolkit (NXT).[30] While graph-based query languages arguably offer the most flexible and powerful types of searches, they are also computationally expensive. Therefore, current implementations are limited to corpus sizes far below those of typical Web corpora.

The four approaches also differ in the type of results they return (the *query matches*). Boolean searches return matching documents or sets of tokens (i.e., instances of the search terms in each document). Regular query languages return contiguous sequences of words that match the specified lexico-syntactic pattern, and most implementations allow individual tokens within the sequence to be marked as "targets". Context-free grammars also return contiguous strings, but will often indicate substrings that correspond to constituents of the grammar (i.e., left-hand sides of the context-free rules), leading to a more structured search result. Finally, graph-based query tools return arbitrary tuples of graph nodes, which will often mix surface tokens with annotation nodes.

We consider regular query languages the most useful choice for searching Web corpora, because they strike a good balance between expressiveness and efficient implementation. Although a more structured representation of query results would sometimes be desirable, large result sets can only be stored and manipulated efficiently when they are limited to contiguous sequences (which

[28] `http://tedlab.mit.edu/~dr/Tgrep2/`

[29] `http://www.ims.uni-stuttgart.de/projekte/TIGER/TIGERSearch/`

[30] `http://nite.sourceforge.net/`

can compactly be represented by their start and end positions). Both the IMS Corpus WorkBench and Manatee seem to provide a good starting point for the implementation of the WaCky query engine. Being open-source software, they can be modified and extended to meet the requirements formulated in section 4.1. Since Manatee is closely modeled on CQP, we will henceforth use the label CQP (or, more generally, Corpus WorkBench) to refer collectively to both tools. There is only very limited information available on the query language and performance of Xaira at the moment, hence we do not pursue this option any further.

4.3 Indexing and compression

For fast online queries, *indexing* of the corpus and its annotations is essential in order to avoid linear search of the entire corpus data, which will typically occupy many gigabytes of disk space (even up to the terabyte range for broad crawls of the Web, cf. Clarke et al. 2002). In the most common form, an index contains, for every word and every annotation value, a complete ordered list of the occurrences of the word or annotation in the corpus.[31]

With the help of an index, simple keyword queries can be performed by direct lookup rather than by linear search through the full corpus. Especially for low frequency terms, index-based search can thus be faster by several orders of magnitude in some cases. More complex queries can still make use of the index by narrowing down the search space to stretches of text that contain an occurrence of the least frequent term specified: a query for noun phrases ending in the noun *beer* only has to consider sentences (or even

[31] Web search engines can substantially reduce the data size of their index by removing high frequency "stop words", which are then ignored in user queries. While this approach makes sense in an information retrieval setting (the word *the* is not a good indicator of the topic of a given document, since it occurs almost everywhere), stop words will often play a central role in linguistic searches and cannot be removed from the index.

smaller units) in which *beer* occurs at least once.

Index-based optimization of regular queries is often problematic because of their sequential, "left-to-right" structure. If the first term of a query is relatively infrequent, its occurrences can be looked up in the index. Matching of the regular expression pattern is then attempted only starting from these positions rather than from every token in the corpus. Such a strategy is employed by the CQP query processor, but it fails whenever the first query term describes a high frequency category such as a determiner or a noun.[32] Using an infrequent term that occurs in another place in the query for index lookup poses technical challenges. Because of the complex patterns of alternatives, optionality and repetition allowed by a regular expression, it is non-trivial to determine whether a given term must necessarily be part of every match (only in this case can it be used for index lookup). Even when such a term has been found, it will not be clear how far the start position of a match might be away from an occurrence of the term, so that the regular expression has to be matched "inside-out" from the lookup term rather than in the common "left-to-right" fashion.

Index-based optimization fails completely for regular queries that search for sequences of POS tags or other very general and frequent categories, e.g., queries that function as a "local grammar" for a particular syntactic construction. In this case, index lookup will have no substantial benefit, even if the final result set is very small. Optimization of such purely "grammatical" queries would only be possible with an *extended index* that includes combinations of POS tags in various configurations, combinations of POS tags with lexical elements, and perhaps also combinations of high frequency words. However, there is no limit to the number of relations that might need to be indexed: pairs of nearby POS

[32]The problem is compounded by the fact that there may be multiple potential start positions for a match in a regular expression, if the expression begins with optional elements or with a disjunction of alternatives.

tags at different levels of maximal distance, combinations of three or more POS tags, etc. Comprehensive indexing would thus lead to an explosive growth of data size beyond all practical limits.

Even in those cases where the index lookup narrows down the search space drastically, the resulting performance gain will often not be as large as one might have hoped. The reason for this behavior is that occurrences of the lookup term are usually spread evenly across the corpus, so that matching the full regular expression query requires random access to the huge amount of corpus data on disk. Purely index-based queries can be processed more efficiently because they access data sequentially, reducing the number of disk seeks and data blocks that have to be loaded. Such index-based implementations are straightforward and widely used for Boolean queries. This is what makes search engines like Google as fast as they are, and it may also be the key to fast online searches through Web corpora. Both CQP and Manatee provide at least rudimentary functionality for Boolean queries, though this feature does not fit in well with their standard regular query languages.

A final topic to be addressed in this section is the issue of data compression. Since disk reads are comparatively slow even when the data are accessed sequentially, as much of the corpus data as possible should be cached in main memory (where random access also is much less detrimental than for on-disk data). Therefore, better data compression translates directly into faster query execution: the benefits of a compact representation easily outweigh the decompression overhead. The IMS Corpus WorkBench applies specialized data compression techniques (Witten et al., 1999) both to the corpus data (word forms and their annotations) and to the index files.[33] Aggressive data compression is not without draw-

[33] In this way, the disk size of a 100-million word corpus (including the index, but without annotations) can be reduced to approximately 300 megabytes. For comparison, a plain text version of the same corpus has a size of 500 megabytes, and a gzip-compressed version has a size of 175 megabytes (but note that these

backs, though, mostly with respect to query execution speed. The block compression technique used by the CWB to store sequences of word forms and annotations makes random corpus access expensive even when the data are cached in main memory.[34]

To summarize the main points of this section, we have seen that indexing is essential to process online queries fast enough for interactive sessions. While basic indexing is a well-understood technique, it is of limited use for most linguistically interesting queries. Clearly, further research into suitable indexing techniques is needed in order to develop a powerful and fast query engine for Web corpora. The usefulness of data compression techniques is debatable, provided that fast and large hard disks are available. A stronger focus on extended indexes may be called for, not least because compression has fewer drawbacks for index data than for the text itself and its annotations.

4.4 The corpus as Web

While a powerful query language and a fast query processor are certainly essential for the linguistic exploitation of Web corpora, there are other important requirements as well. The potentially huge result sets returned by a query have to be managed and presented to the user, a task for which query engines like CQP and Manatee provide only rudimentary functionality. A minimum requirement is that users must be able to browse the query results (displayed with varying amounts of context), sort the matches according to different criteria, and look at random subsets of the results to get a broad overview. Especially for very large sets of results, additional functionality is desirable that helps to reduce and structure the massive amounts of data brought up by the

sizes do *not* include any index data).

[34] The reason is that for every access, an entire block of data (usually 256 tokens or more) containing the relevant token has to be decompressed.

corpus query. For instance, it should be possible to compute frequency lists of the matching word sequences (or individual target elements), to calculate distributions of the matches across metadata categories, and to identify collocations (in the sense of Sinclair 1991) or collostructions (Stefanowitsch & Gries 2003). All these functions are provided by BNCweb, a user-friendly interface to the British National Corpus (see, e.g., Hoffmann and Evert 2006).

While such analyses can be performed online for moderately large result sets, more advanced analysis options (e.g., exhaustive collocational analyses of the lexico-syntactic behavior of a word and automatic identification of other terms and phrases that have a similar distribution in the corpus) would further increase the usefulness of Web corpus data. Such complex analysis functions can only be performed offline, and the same is true for simpler functions when they are applied to result sets that contain millions of matches.

For each type of analysis, the final results can be represented as a table of corpus frequencies, statistical coefficients, similarity measures, etc. (usually linked back to individual query matches). A *relational database* software is ideally suited to store, process and query such tabular data structures (and this is the approach that BNCweb takes). Such a database provides an excellent environment to combine results from online and offline processing, where the latter can either stem from offline analysis of query results or from precompiled frequency tables for common words and phrases. We recommend the open-source implementation MySQL,[35] which is widely acclaimed for its stability, speed and flexibility.

A sketch of an architecture for the WaCky search engine is beginning to take shape, but we have to deal now with at least three distinct software packages: the query engine proper, a result browser, and a relational database. Moreover, at least two of these

[35] `http://dev.mysql.com/downloads`

tools require some amount of practice and in-depth knowledge of their (not entirely intuitive) query languages in order to achieve good results. Does this mean that Web corpora are essentially inaccessible for novice and non-technical users?

The enormous popularity that Google enjoys among linguists can only in part be explained by the fact that it makes an unprecedented amount of language data available. We believe that an equally important role is played by the fact that Google search is easy to use and can be accessed through a familiar user interface, presents results in a clear and tidy way, and that no installation procedure is necessary. For these reasons, we conjecture that the success of the WaCky query engine and its acceptance among linguists will hinge on its ability to offer a similarly user-friendly, intuitive and familiar interface. As in the case of Google, a Web interface has the potential to satisfy all three criteria. In other words, we should not only use the Web as a corpus, but also present the *corpus as Web*, i.e., provide access to Web corpora in the style of a Web search engine. A crucial advantage of the "corpus as Web" approach is that it allows us to hide the three (or even more) quite different components of the WaCky query engine behind a uniform Web interface. For the end user, the transition between query engine, result browser and tables in a frequency database will be seamless and unnoticeable, even if the technical implementation of this integration is a complex task. The key insight here is that complexity can and must be hidden from the user. Once again, BNCweb provides a good illustration of this approach, and a substantial part of the functionality sketched here has been implemented in the commercial Sketch Engine (built on top of Manatee and MySQL).

What is most urgently needed by the community now is an open-source implementation of a "corpus as Web" framework for the WaCky query engine, which should be easily configurable and extensible with new modules (providing, e.g., alternative visualiza-

tions of query results, additional analysis functions, or simplified query languages that shorten the learning curve for new users). For individual components of the system, open-source software packages are already available (such as the IMS Corpus WorkBench, Manatee and MySQL, as well as specialized software packages for statistical and distributional analyses), but may need to be improved and extended in order to meet the requirements listed in section 4.1. We are currently working on a detailed sketch of a possible architecture for the WaCky query engine and suggestions for the implementation of its components.

5 Conclusion

This introductory article looked at different ways in which the by now ubiquitous expression *Web as Corpus* can be interpreted, and provided an overview of the major issues involved in turning WaC from hype to reality. While doing this, we tried to provide a survey of some recurring themes in this collection, as well as describing some of our current and future work.

Despite the many daunting tasks that we might encounter on the way to its exploitation (actually, in part *because* of these daunting tasks), the Web is probably the most exciting thing that happened to data-intensive linguistics since the invention of the computer, and we would like to conclude this introduction by reiterating our invitation to the readers to engage, with us, in the WaCky adventure.

References

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. *Proceedings of LREC 2004*, 1313-1316.

Baroni, M. and Kilgarriff, A. (2006). Large linguistically-processed Web corpora for multiple languages. *Proceedings of EACL 2006, demo session*, 87-90.

Baroni, M. and Ueyama, M. (2006). Building general- and special-purpose corpora by Web crawling. *Proceedings of the 13th NIJL International Symposium*, 31-40.

Baroni, M., Kilgarriff, A., Pomikálek, J., Rychlý, P. (2006). WebBootCaT: Instant domain-specific corpora to support human translators. *Proceedings of EAMT 2006*, 247-252.

Broder, A., Glassman S., Manasse, M. and Zweig, M. (1997). Syntactic clustering of the Web. *Proceedings of the Sixth International World-Wide Web Conference*

Chakrabarti, S. (2002). *Mining the Web: Discovering knowledge from hypertext data*, San Francisco: Morgan Kaufmann.

Chakrabarti, S., Roy, S. and Soundalgekar, M. (2002). Fast and accurate text classification via multiple linear discriminant projections. *VLDB Journal* 12(2), 170-185.

Chklovski, T. and Pantel, P. (2004). VerbOcean: Mining the Web for fine-grained semantic verb relations. *Proceedings of EMNLP-04*.

Clarke, C. L. A., Cormack, G. V., Laszlo, M., Lynam, T. R. and Terra, E. L. (2002). The impact of corpus size on question answering performance. *Proceedings of SIGIR '02)*.

Hoffmann, S. and Evert, S. (2006). BNCweb (CQP-edition): The marriage of two corpus tools. In Braun, S., Kohn, K. and Mukherjee, J. (eds.) *Corpus technology and language pedagogy: New resources, new tools, new methods*, Frankfurt am Main: Peter Lang, 177-195.

Jurafsky, D. and Martin, J. (2000). *Speech and language processing*, Upper Saddle River: Prentice Hall.

Kilgarriff, A. and Grefenstette, G. (2003). Introduction to the special issue on the Web as corpus. *Computational Linguistics* 29(3), 333-347.

Manning, Ch. and Schütze, H. (1999). *Foundations of statistical natural language processing*, Boston: MIT Press.

Schiehlen, M. (2004). Annotation strategies for probabilistic parsing in German. In *Proceedings of COLING 2004*, 390-396.

Sinclair, J. (1991). *Corpus, concordance, collocation*, Oxford, OUP.

Stefanowitsch, A. and Gries, S. (2003). Collostructions: Investigating the interaction of words and constructions. *International Journal of Corpus Linguistics*, 8(2), 209-243.

Varantola, K. (2003). Translators and disposable corpora. In Zanettin, F., Bernardini, S. and Stewart, D. (eds.) *Corpora in translator education*, Manchester: St. Jerome, 379-388.

Witten, I. H., Moffat, A., and Bell, T. C. (1999). *Managing gigabytes, 2nd edition*, San Francisco: Morgan Kaufmann.

Experience Building a Large Corpus for Chinese Lexicon Construction

Thomas Emerson and John O'Neil

1 Introduction

The World Wide Web (WWW) provides a large and constantly growing renewable source of natural language data in many of the world's languages. Computational linguists and lexicographers have been trying to harness this bounty (and arguing about its applicability to any given task) for over six years (Kilgarriff and Grefenstette 2003). This chapter discusses our experience with using the Chinese Web for lexicon construction, focusing on the low-level details and problems we experienced during our initial proof-of-concept experiments.

1.1 Chinese text segmentation

Chinese is written without the use of spaces between words, which is problematic for natural language processing (NLP) applications which operate on words, information retrieval and data mining being two important and lucrative examples. The importance of accurately segmenting Chinese has made it an area of active research (Sproat and Emerson 2003; Wu 2003; Gao et al. 2004) around the world, and a variety of methods are used.

Initial attempts at addressing the problem used a variety of dictionary-based methods, such as maximal-matching: starting from the beginning of each sentence find the longest match in a dictionary, and move forward until the sentence is exhausted.

If a multi-character word cannot be found, then it is treated as a single character word and we move to the next position. Modifications of this simple algorithm to account for word frequency and other heuristics (such as POS compatibility) have been proposed. These techniques are often only as good as the dictionary that supports them, and as is well known constructing a lexicon is a time consuming (and often expensive) proposition. Further, unless the dictionaries are regularly updated they soon become stale as new words are used.

The lexical approach to word segmentation was overshadowed by the use of various statistical methods. These systems can be quite effective, so long as the text being processed is similar to that used to train them. These methods also suffer from a severe resource bottleneck problem, since they *a priori* require segmented text, which is (like lexicons) time consuming and expensive to obtain.

Recently more hybrid approaches have been proposed that utilize a mixture of statistical and lexical information. While these systems would seem to mitigate each other's limitations, they still need a comprehensive lexicon.

Building an electronic Chinese lexicon for use in a segmentation system is problematic: to generate the dictionary you need to segment a text collection, but to segment the text collection you need the lexicon. Substantial work has been done on developing techniques for constructing lexica with little or no human supervision from unsegmented text (Ge et al. 1999; Chang and Su 1997; Lin and Yu 2004; Jin and Wong 2002). In all cases the various techniques require a significant corpus to work on, due to the Zipfian nature of word frequency distributions (Baayen 2001).

1.2 Using corpora for linguistic analysis

Collecting a large corpus of Chinese text is challenging and difficult on its own, but of course the purpose of corpus collection is to

put it to good use. Extracting information about Chinese from a corpus poses a number of unusual challenges, and it is illuminating to discuss them, especially in comparison to corpus work in other languages.

Possibly the most common task for which corpora are used is lexical development, and this is precisely where the first difficulty lies. Because Chinese words are written without interstitial white space, it is necessary to develop a tokenizer for Chinese before one can develop a dictionary from a corpus. However, since the most obvious ways to tokenize arbitrary Chinese text involve using a lexicon, we immediately have a chicken-and-egg problem.

In practice, it has almost always been easier to use an existing lexicon to form the core of a tokenization tool, since lexica are more common than large tokenized corpora. (Of course, both can be used together, leading to systems that employ both lexical and statistical knowledge for segmentation.) In the absence of a good corpus, or in the absence of a corpus relevant to the domain of interest, and with a lack of relevant training data, a purely corpus-driven, unsupervised approach must be chosen.

Work on Chinese segmentation using only corpus materials has been an active research topic for some time (Sproat et al. 1994; Sun et al. 1998), and is still active (Gao et al. 2004). In general, abstracting away from statistical details, these methods look for sequences of characters that occur together more often than expected, and the more often they co-occur, the more likely they are to form a single token. This takes place in the context of an assumed segmentation algorithm. Any reasonable segmentation algorithm has to balance the absolute likelihood of a token, the likelihood of a token in a context, the total likelihood of a proposed segmentation of an entire sentence, and the possibility that there might be a new word in the sentence. These choices affect what lexical items are found, and how sentences are segmented.

Because these choices all have effects on the statistical anal-

ysis of the corpus, it can be difficult to create a consistent segmentation standard across a corpus, using entirely unsupervised methods. Since there is no generally accepted definition of what constitutes a word in Chinese, it can be unclear for an unsupervised learner, as it is for a human, how to decide on an appropriate level of granularity for segmentation, and apply it consistently. A lexicon is not created in a vacuum, and it can be difficult using unsupervised learning to create a lexicon useful for tasks other than segmentation, such as POS tagging.

Given the volume of the data, it would be advantageous to make the segmentation learner work incrementally on a stream of documents. Most previous work assumes a static, though large, corpus. Nevertheless, a continuous stream of new documents allows more accurate segmentation to be created over time. Also, if an unsupervised segmentation learner works on a continuous stream of documents, it has the benefit that it can be extended to continuously find new lexical entries. This is especially important in Chinese, since most new words permit alternate segmentations using tokens already in the corpus. Only additional statistical information, especially on new documents which might have a "burst" of a new word, can help identify new words.

There are other types of unsupervised learning from a corpus, especially once there is a reliable segmentation for the corpus. For example, clustering tokens based on their neighbors can bootstrap an assignment of POS tags to tokens in a lexicon. Clustering can also be used to group documents based on similar bags of words. However, since the lexical data is sparse, some clustering algorithms may not be optimal. In fact, since Zipf's law holds for even the largest of corpora, we are assured of continued sparseness.

As the corpus grows, we gain increased accuracy at the cost of being forced to process ever larger amounts of information. This makes it necessary to tune learning algorithms to use large corpora. The most common way to do this is to implement learning that

increases the precision at the expense of recall (Pasca 2004). As the corpus grows larger, it matters less what might be missed, since it will be seen again and again, but it becomes more important to avoid learning the noise and choking on the collected corpus data.

2 Problem statement

Crawling and post-processing large amounts of Chinese-language data are the first steps in a system designed to perform nearly continuous lexical development. The ultimate goal of the project is to develop an environment for finding and tagging possible neologisms in text from all Chinese-speaking communities for human adjudication before inclusion in a lexicon. This is similar to the ongoing LIVAC (Linguistic Variation in Chinese Communities) project at the City University of Hong Kong (T'sou et al. 1997). Unlike LIVAC, which focuses on news sources, we are interested in casting as wide a net as possible to include data from numerous registers and language levels.

We are also interested in utilizing the data we collect for other non-lexicographic purposes, such as the construction of POS tagged and entity tagged corpora for other high-level NLP tasks. These activities are currently secondary to the primary goal of lexicon construction.

Given the usage requirements, the following desiderata are apparent:

1. A Web crawler capable of processing millions or tens of millions of URIs (Uniform Resource Identifiers): a crawl will start with a seed set of several thousand URIs and will discover thousands more as it progresses.

2. The crawler must be polite to the sites being crawled, while optimizing throughput. This means obeying a site's robot-

exclusion preferences and not fetching documents from the site multiple times per second.

3. We are only interested in textual information, so we want to avoid downloading images, sounds, movies, and other arbitrary binary content: this is both a waste of bandwidth and storage. Ideally we would only download Chinese language content.

4. We do not want to wait for the crawl to complete before starting to process the data: this will become more important as the amount of stored text increases.

5. We want to be able to regularly recrawl sites that are known to change on a regular basis: online news sites and personal Web logs (blogs) are two obvious examples.

Writing a robust Web crawler from scratch would be an interesting project, and doing so may be a feasible solution for building small corpora. There are source code libraries available that provide HTTP protocol support and that can extract outgoing links from HTML, and these can be used in any number of programming languages including Python, Perl, and C/C++. Indeed, for a small enough collection just using a text-mode browser like `lynx` may be sufficient. However, for crawls on the scale we foresee the thought of having to maintain and extend the crawler was unappealing.

There are some open-source, command-line based downloading tools, such as `wget`[1] which can be used for downloading content, and even for mirroring entire sites. However, it has often been misused to the point that many sites' `robots.txt` file blocks all access from it. `wget` is also not designed to be a crawler and not suited for large ongoing crawls. We experimented with a modified

[1] `http://www.gnu.org/software/wget/wget.html`

version of `wget` for website archiving and link analysis but found that more time was spent inside the code than was spent on the real task. This is not to say that `wget` is poorly written, rather it simply did not serve our exact needs.

Ubicrawler (Boldi et al. 2004) has many of the features we could hope for, based on the above desiderata. However, it is not publicly available and we were wary of using software whose source we did not have ready access to.

Heritrix (Mohr et al. 2004) came to our attention when it was first announced in January 2004. We began using it for small focused crawls starting in May 2004, contributing bug fixes and new functionality to better serve our (and hopefully other linguistic researchers') needs. We quickly gathered experience with the code base, the developers, and the architecture and found it to fit our requirements very well.

3 Heritrix overview

Heritrix is an open-source Web crawler developed by the Internet Archive (IA).[2] Development of the crawler began at the beginning of 2003 after they determined that it would be beneficial for them to perform crawls internally. The crawler is written in Java, is modular, multi-threaded, and is capable of handling large crawls: the National University of Iceland has used it to crawl the entire `.is` domain (11,000 domains, 35 million URIs).[3]

There are three primary interacting components in Heritrix: the *Scope*, the *Frontier*, and the *Processor chains*.

The Scope determines whether or not a discovered URI should be included in the crawl, without actually fetching the data pointed to. Scopes can limit URIs to certain domains or sub-domains. A scope can use arbitrarily complex regular expressions to make the

[2] `http://crawler.archive.org/`

[3] `http://groups.yahoo.com/group/archive-crawler/message/1385`

decision, and can reject files that are more than a certain number of links from a seed URI. Users can also develop their own scopes in Java if the built-in modules are inappropriate for a particular application.

The Frontier maintains the internal state of the crawl. It keeps track of which URIs have already been fetched, which are scheduled to be downloaded (i.e., that have been declared in scope), and which are currently being processed. It is responsible for determining which URI should be fetched next, paying attention to limitations set by a site's `robots.txt` file and to other forms of "politeness".

The Processor chains contain modules that operate over the URIs (and associated data, once it is fetched) to perform actions ranging from URI normalization to filtering based on length or headers to writing the fetched data to disk and providing crawl status information. Much of Heritrix's power lies in the configurability of these processor chains. All of the processors, as well as the scopes and frontiers, are extremely configurable.

Items downloaded by Heritrix are stored in an "ARC" (Web archive) file,[4] along with associated metadata, including the original URI, time stamp of when it was downloaded, MIME header, length, and fingerprint. By default each ARC file contains up to 100 MB of compressed data: during the crawl Heritrix maintains a pool of open ARC files (signified on disk by the ".open" extension on their file name) into which content is written by the crawler as it is processed. When an ARC file is full, the `.open` extension is removed and that ARC file is "complete": it will not be touched by the crawler again. This makes it possible to work with a crawl's ARC while the crawl continues to run – it can be moved to more permanent storage or its contents processed immediately.

Heritrix was designed for synchronic archiving and does not

[4]The format for ARC files is available at `http://www.archive.org/Web/researcher/ArcFileFormat.php`

support incremental crawling: an incremental crawler will refetch pages on a regular basis and update the stored copy with the updated version if it is different. Nevertheless, the ability to revisit sites was added to Heritrix (Sigurdsson 2005). At the time of writing, we have not had an opportunity to evaluate this addition.

The crawler has a Web-based user interface (WUI) that allows you to setup and monitor crawl jobs. You can define profiles with common settings that can be reused. These are stored on disk in XML and can be edited (or created) outside of the UI. You have full control of every aspect of the crawl operation from this console. Recent versions of the software have added a JMX (Java Management Extensions) interface, allowing it to be controlled from any JMX-enabled application or device.

Heritrix has an active developer community. The core team at the Internet Archive is supplemented by a number of people from around the world in both industry and academia, including linguists, digital librarians, and computer scientists. Further, they have worked with the Ubicrawler developers to incorporate some of their code.

3.1 Heritrix vs. desiderata

Successive generations of Heritrix have become increasingly capable in performing large crawls. While the Icelandic crawl mentioned in the previous section was done over 4 separate crawl jobs, it is believed that the entire 35 million URI snapshot could be done with a single crawl job. This is more than sufficient for our first requirement.

The default configuration for the crawler is to look for and obey a site's `robots.txt` file: this is an inviolate prerequisite that the users need to go out of their way to circumvent. There are numerous configurable settings for throttling the frequency with which documents are fetched from a given server. For example, you can set the delay between successive requests as a function of

the round-trip time of the last request made to a server. The frontier can also be configured with different scheduling mechanisms for handing off URIs to the worker (or toe, in Heritrix parlance) threads. Therefore only through operator error (or malice) will the crawler be impolite.

Given that one of the express design goals of Heritrix is to archive the Web, it is no surprise that in its default configuration it will attempt to fetch *everything* it can (assuming it isn't prevented by the `robots.txt`, of course.) However, through the use of the existing filtering mechanism offered by the architecture, one can almost eliminate all unwanted data from the crawl.

Content fetched from the Web is written into ARC files, which are closed after they reach around 100 MB in size. From that point Heritrix is done with them and they are available for processing: one can develop a work-flow that starts operating on the data while the rest of the crawl continues on.

Incremental crawling is the only desideratum that the current release of the crawler lacks, though Sigurdsson's (2005) work looks promising. For our current needs incrementality is not essential: we can just start new crawls based on the previous ones, relying on post-processing to remove duplicate documents.

4 Practical crawling issues

4.1 Seed generation

Our goal is to collect as much text as possible: rather than looking for specific linguistic constructs we need vast amounts of text to mine for neologisms. To this end we needed some way to find thousands of URIs with which to seed our crawls.

The Open Directory Project (ODP)[5] claims to be, "the largest, most comprehensive human-edited directory of the Web." Hun-

[5] `http://dmoz.org`

dreds of volunteers world-wide categorize millions of URIs according to defined criteria. All of the ODP data is freely available under the Open Directory License, which allows unlimited research and commercial use of the data as long as appropriate attributions are made and the rights outlined in the license are not impinged by subsequent distribution. Snapshots of the ODP database are made in slightly modified RDF every month or so: the release dated 28 July 2005 was 210 MB compressed and 1 GB uncompressed, containing some 4.5 million URIs in 551,578 categories.

The classification scheme used by the ODP includes regional and language-specific categories. For example, the category

```
Top:World:Chinese Simplified
```

contains pages that are known to be in Simplified Chinese. There are 1,993 sub-categories of this, counting for 16,535 URIs. The upshot of this is that it is trivial to extract all URIs in this category from the RDF file.

The ODP data is processed by extracting the categories and associated URIs from the RDF into a simple two-column tab-delimited file containing just the category and the URI. This reduces the size of the ODP database by almost 50% by eliminating the XML markup and removing unused information. This only has to be done once for each release of the data. After this file is created, it is trivial to extract just the links matching a particular category using `grep` and `cut`:

```
% grep Top/World/Chinese_Simplified ext.ut8 |
     cut -f 2 > zh_sc_uris.txt
```

A simple Python script is then used to generate a random sample of the extracted URIs:

```
% python pick_random.py 1500 zh_sc_uris.txt > seeds.txt
```

The resulting `seeds.txt` can now be used in a Heritrix job specification.

4.2 Job configuration

For lexicon construction we are only interested in HTML documents. Other document types, such as PDF or Microsoft Word, require more extensive processing than we chose to deal with. The first approximation for this is to exclude URIs from the scope with file extensions we do not care about. This can be done with a `URIRegExpFilter` with a long regular expression similar to:

```
.*(?i)\.(gif|pdf|wav|dvi|ps|iso)$
```

The expression that we actually use is considerably larger, containing 176 extensions.[6] There are two file extensions that could not be included in the filter, `au` and `txt`. The `au` cannot be excluded because this would cause sites in Australia (whose ISO 3166 code is also `au`) to be excluded from the scope in some situations, and `txt` was kept because its omission would cause `robots.txt` to be excluded, violating the hard prerequisite Heritrix has for handling the robot exclusion protocol.

Unfortunately filtering just on file extension does not exclude all content: very often URIs that yield images or PDFs are generated from CGI scripts or other dynamic methods and lack a file extension. To account for these cases, we install a `ContentType RegExpFilter` as a MidFetch filter (run after the HTTP response headers are received but before the content) to filter on the content type:

```
(?i)text/html.*
```

There are two other options that need to be set for each job. The first is `default-encoding`, which is the character encoding that is used for files that do not explicitly declare one. When working with multi-byte character sets it is important that Heritrix

[6] `http://www.dreamersrealm.net/~tree/blog/?p=4`

know what encodings it is likely to see. Failure to set this appropriately can result in broken link extraction. The second is to add an appropriate `Accept-Language` header to the `accept-headers`. Some sites do content negotiation to send appropriately translated content to the browser. Without explicitly specifying the `Accept-Language` you may not receive the content you expect. For Simplified Chinese sites it is best to set the default encoding to CP936 (Microsoft's Simplified Chinese code page) and add:

```
Accept-Language: zh-cn, zh-sg
```

For Traditional Chinese sites, the default encoding is CP950 (Microsoft's Traditional Chinese code page) and the accept header:

```
Accept-Language: zh-tw, zh-hk
```

5 Crawl experiences

Using the methods described in the previous section we generated a random set of 1,500 Simplified Chinese URIs from the May 2005 ODP data release. A sample of the 16,000 URIs available in the ODP Simplified Chinese section were used to constrain the size of this crawl. We ran a local pre-release build of Heritrix 1.4.0 on an old dual-CPU 666 MHz x86 machine with 1 GB physical memory and running Gentoo Linux 2005.1 with Sun's JDK 1.4.2. This machine was dedicated to the crawl. We gave the Java virtual machine a 512 MB heap and this was sufficient for the crawl.

The crawler was initially configured to use 50 threads (i.e., fifty concurrent connections). This was increased every other day until we reached 150 threads. We elected to use the "Domain" scope, which allows any URI in the domain of one of the seeds to be crawled. A depth restriction (number of hops from a seed) of 25 was used. We let the crawl run for approximately 11 days before manually stopping it due to a lack of disk space. When

URIs stored:	7,372,351
ARC files:	300
Total ARC File Size:	28 GB
Unique Hosts Crawled:	4,032
Total HTML size:	109.7 GB
Total Stripped size:	15.8 GB
Languages found:	28

Table 1. Statistics on the first large Chinese crawl

Simplified Chinese	5,510,748	Romanian	52
Traditional Chinese	50,030	Persian	38
Russian	5,986	Hungarian	32
Japanese	4,059	Finnish	28
Korean	393	Bulgarian	26
Arabic	365	Spanish	11
Polish	198	Albanian	11
Greek	136	Vietnamese	10
Thai	120	Swedish	8
Turkish	83	Latvian	5
Czech	67	German	5
Portuguese	65	Icelandic	3
Hebrew	58	Slovak	2
Lithuanian	55	French	1

Table 2. Breakdown of languages found in the first large Chinese crawl

the crawler was shutdown it had stored 7,372,351 URIs, or approximately 27,926 per hour, or around 8 documents per second. Further statistics on the crawl can be found in tables 1 and 2.

5.1 Disk issues

The gating factor on the length of this crawl was disk space: the crawl had been running for almost two weeks until running out of disk space due in large part to the amount of "state" data that was being stored: it dwarfed the amount of data stored in the ARC

files (48 GB to 28 GB). This saved state data is only needed during the crawl: once the crawl is terminated the state information can be deleted. It appears that the ratio of state to "content" is highly dependent on the type of content being stored: our use of Heritrix to only download textual data is somewhat unique. The IA has observed that for archival crawls the state is only around 15% of the ARC file size.[7] The Heritrix developers were subsequently able to implement some size reduction on the data stored in the state files, though we have not had an opportunity to study the effects of this change in our crawls.

During a crawl "disk contention" can become a performance bottleneck too:

- The crawler keeps a pool of ARC writers, which the threads use to write the content they are downloading. Each of these contends for the disk. Interestingly enough, the IA found that increasing the number of ARC writers does not help performance, but can actually lower it. The rationale is that increasing the number of writers increases the amount of contention for the disk, which ends up being a more time-consuming operation than keeping threads waiting for a writer.

- The crawler maintains at least four log files during the crawl, so there is contention for writing (and, for the administrative interface, reading) these.

- The state data. As observed with the current crawl, there is a lot of this: not only does it consume disk space, it can result in disk contention during the crawl.

Heritrix allows you to split the ARCs, logs, and state across different physical disks. This can go a long way to reducing con-

[7] `http://groups.yahoo.com/group/archive-crawler/message/1870`

tention on a single disk, and is the recommended way of dealing with this.

Based on our experience with the Chinese crawl, we need to allocate about 150% of the space taken by the expected crawled data size for state information. This storage is only needed during the crawl, and can be reclaimed when it completes. This becomes a real problem if we run multiple large-scale crawls on a single machine where you could expect to use 50-150 GB of disk space, per crawl, for the state information.

6 Post processing

For vocabulary acquisition we need to extract the raw text from the Chinese documents stored in the ARC files. This processing was done after the crawl was completed, but it could be done incrementally as the ARC files are closed: the steps are repeated for each ARC.

Post-processing is done in two phases: we first extract all interesting documents from the ARC files, and then lift the text from the HTML.

The first phase works as follows: each `text/html` item in the ARC file has its HTML markup stripped. If the amount of text left after removing markup is greater than a threshold (1024 bytes) then we perform language and encoding detection using Basis Technology's Rosette Language Identifier, a commercial language/encoding detection system. With the 1024 byte threshold the identifier is almost 100% accurate. Documents that are detected to be Simplified Chinese (regardless of character encoding) are then marked for further processing.

Items that reach this stage are extracted from the ARC file in the original HTML, *except* that they are transcoded from the detected character encoding to UTF-8 with HTML character entities expanded. These are written to disk into numbered files

contained in numbered directories, with at most 1,000 files per directory. This is done since few file systems are capable of working reliably with directories containing tens of thousands (if not millions) of files. Note that we do not rewrite any character set declarations that may exist in the original HTML file: these are never used.

Once the ARC files have been processed in this way, they can be moved to offline storage since the "interesting" content has been extracted. Our policy is to keep the ARCs for each crawl for future use and research.

The second post-processing phase is lifting the text from the markup. For some purposes (though not necessarily lexicon extraction) it is useful to have the rough physical structure of the text preserved, and many utilities which merely remove markup do not preserve this. We envision "sifting away" the markup and leaving the text in place, with structure preserved. To do this we use an open source tool called `vilistextum`[8] which is robust in the face of "broken" markup and does a decent job of preserving the logical structure of the documents. Each file extracted in the first phase is passed through Vilistextum and saved with the same basename but different file extension. The HTML files generated in phase 1 can then be deleted (since they can be trivially regenerated from the ARCs) or moved to offline storage. Table 3 gives some statistics on the resulting text.

We do not yet perform any (near-)duplicate or boiler-plate removal. This is an important future direction, and we are examining various techniques to do this. Most existing duplicate document detection algorithms presume efficient tokenization of the input documents, which we do not have in the case of Chinese. This is a problem that we will need to tackle in the near term, since duplicate documents will artificially inflate the statistics we use to find new words in the texts.

[8] `http://bhaak.dyndns.org/vilistextum/`

Number of files:	3,291,985
Average file size:	4,935 bytes
Total *hanzi*:	3,861,758,249

Table 3. Statistics on the Simplified Chinese text

7 Data management

7.1 Crawl data

Data management becomes a significant issue as the size of the crawls increases.

Given that the content we store is almost exclusively textual the compression ratios are quite good (17:1). However, a large crawl still generates a lot of data that needs to be stored and backed up.

The raw data that is crawled is not immediately useful for many of our tasks, so it must be post processed. This raises several issues, including: when is the processing performed? Is the processed data saved, or do we always process on demand? How do we deal with the duplicate and near-duplicate data problem? How do we extract the data we're interested in from the huge amount available? Again, compression can be used to help with disk space issues. Do we want (or need) to be able to map back from processed data to the original ARC file and to a specific crawl?

We have no way of knowing how much data is available for a particular set of parameters (e.g., language, encoding, content type).

Backups and data integrity are difficult; backing up 28 GB of ARC files requires at least 7 DVD-R discs. One solution is to use one or more external FireWire drives to archive the data after it is crawled. Unfortunately this single-point of failure caused us to earlier loose about 100 GB of data when the file system on

the external drive became corrupted and unrepairable. This may have been an issue with the FireWire drivers on Gentoo, or an issue with the ext3 file system, or a combination of these.

7.2 Processed data

The data from each physical URI is stored in a single file after all processing is complete. It is possible to work backwards from the file name to the ARC file containing the original HTML. This means, however, that there are hundreds of thousands of files living on the file system, which is obviously problematic for many reasons.

Initially we generated a `bzip2` compressed `tar` file containing the extracted data. We do much of our linguistic processing in the Python language, which has the ability to read the entries of these archives. Unfortunately this didn't work since Python was unable to process files over 4 GB in size (a bug which has since been fixed).

Another large collection, the LDC's Chinese Gigaword (Graff et al. 2005)[9] contains 349 compressed SGML files which in turn contain multiple news articles along with other markup. We could concatenate multiple files into one, and compress this, but doing so involves the addition of extra markup that we do not want to add.

8 Conclusion

Since the crawl documented here, we have performed a second Simplified Chinese and a first Traditional Chinese crawl of similar size. We have not yet started lexicon extraction on any of these corpora, although this will proceed in the near future.

[9] This corpus contains approximately 1.3 billion characters, slightly less than one-third the size of the crawl described here.

Heritrix has worked very well for the tasks we have given it. Nine times out of ten the problems we've encountered have been of our own doing, and the responsive development team have been quick to point out our errors or to correct problems that we have encountered. The software continues to improve, and the architecture is proving itself again and again. The addition of the JMX interface is particularly exciting, as we can envision integrating the crawler into a Web-based lexicographer's workbench.

The biggest concerns are generally pragmatic: finding enough disk space to actually store the crawl data and associated transient state; sharing bandwidth with the rest of the company; post-processing the collected data. These are problems that any large-scale crawling effort will encounter.

Our next steps involve integrating the crawler and its data into the linguistic processing modules of the system, and making the crawls incremental so that we can continue to expand our lexica as time goes on. We are also expanding our crawling efforts into other languages, and looking at ways of expanding Heritrix to perform directed crawls of specific languages for which readily available corpora of any size do not exist (Ghani et al. 2001).

Acknowledgments

The authors would like to thank Michael Stack of the Internet Archive for his comments on the section describing Heritrix. The work reported in this article was conducted while Thomas Emerson worked at Basis Technology.

References

Baayen, R. (2001). *Word frequency distributions*, Berlin: Springer.

Boldi, P., Codenotti, B., Santini, M. and Vigna, S. (2004). Ub-

icrawler: A scalable fully distributed Web crawler. *Software: Practice & Experience* 34(8), 711-726.

Chang, J. and Su, K. (1997). An unsupervised iterative method for Chinese new lexicon extraction. *Computational Linguistics and Chinese Language Processing* 2(2), 97-148.

Gao, J., Li, M., Wu, A. and Huang, C. (2004). Chinese word segmentation: A pragmatic approach. Technical Report MSR-TR-2004-123, Microsoft Research.

Ge, X., Pratt, W. and Smyth, P. (1999). Discovering Chinese words from unsegmented text. *Proceedings of the 22nd International SIGIR Conference*, 271-272.

Ghani, R., Jones, R. and Mladenić, D. (2001). Mining the Web to create minority language corpora. *Proceedings of the 10th International Conference on Information and Knowledge Management*, 279-286.

Graff, D., Chen, K., Kong, J. and Maeda, K. (2005). Chinese Gigaword, second edition. Lexical Data Consortium, LDC2005T14.

Jin, H. and Wong, K. (2002). A Chinese dictionary construction algorithm for information retrieval. *ACM Transactions on Asian Language Information Processing* 1(4), 281-296.

Kilgarriff, A. and Grefenstette, G. (2003). Introduction to the special issue on the Web as corpus. *Computational Linguistics* 29(3), 332-347.

Lin, Y. and Yu, M. (2004). The properties and further applications of Chinese frequent strings. *Computational Linguistics and Chinese Language Processing* 9(1), 113-128

Mohr, G., Kimpton, M., Stack, M. Ranitovic, I. (2004). Introduction to Heritrix, an archival quality Web crawler. *Proceedings of the 4th International Web Archiving Workshop.*

Pasca, M. (2004). Acquisition of categorized named entities for Web search. *Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management (CIKM 04)*, 137-145.

Sigurdsson, K. (2005). Adaptive revisiting in Heritrix. Master's thesis, University of Iceland.

Sproat, R. and Emerson, T. (2003). The first international Chinese word segmentation bakeoff. *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*.

Sproat, R., Shih, C., Gale, W. and Chang, N. (1994). A stochastic finite-state word-segmentation algorithm for Chinese. *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, 66-73.

Sun, M., Shen, D. and T'sou, B. (1998). Chinese word segmentation without using lexicon and hand-crafted training data. *Proceedings of COLING-ACL '98*, 1265-1271.

T'sou, B., Lin, H., Liu, G., Chan, T., Hu, J., Chew, C. and Tse, J. (1997). A synchronous Chinese language corpus from different speech communities: Construction and applications. *Computational Linguistics and Chinese Language Processing* 2(1), 91-104.

Wu, A. (2003). Customizable segmentation of morphologically derived words in Chinese. *Computational Linguistics and Chinese Language Processing* 8(1), 1-28.

Creating General-Purpose Corpora Using Automated Search Engine Queries

Serge Sharoff

1 Introduction

The Internet is a natural source of linguistic data, providing an abundance of texts of various types in a large number of languages. These texts are already in electronic form suitable for corpus studies, either as downloadable pages, or as a resource to be searched using search engines. On the other hand, large representative corpora of the size of the British National Corpus (BNC, Aston and Burnard 1998) exist for very few languages, because they are expensive to build. They are absent even for major world languages, such as Chinese or French. Many ad-hoc text collections are available, but they are restricted in either their size or the variety of text types. Typically they are produced on the basis of out of copyright fiction (such as Project Gutenberg)[1] or newswire/newspaper texts that are available in large quantities and relatively easy to acquire from their publishers (e.g., the Reuters corpus for English (Rose et al. 2002), or the Gigaword corpora for Arabic, Chinese and English (Cieri and Liberman 2002). News corpora are useful for many applications, such as development of gazeteers, parsing and word sense disambiguation algorithms, yet they cannot replace corpora representative of general language, such as the BNC, as

[1] `http://www.gutenberg.org/`

the former reflect only the formal register of reporting news stories, while corpora that are claimed to be representative should include a variety of text types. Below we compare the language of news corpora against the language used in the BNC and the language derived from the Web. The comparison shows that the news corpora differ significantly from either representative or Internet corpora and cannot provide a window into modern language use in general.

The usefulness of Web data is evidenced by numerous corpus studies based on the number of pages returned by Google for specific queries (Kilgarriff and Grefenstette 2003). Some researchers in traditional linguistics also use data returned from Google as the basis for their research, cf. Robb (2003), Volk (2002). However, Google is a poor concordancer. It provides only limited context for results of queries, cannot be used for linguistically complex queries, such as searching for lemmas (as opposed to word forms), restricting the POS or specifying the distance between components in the query in less than crude ways. More importantly results are ordered according to their "relevance" to the topic of the query using page-rank considerations, not according to left or right context as it is often useful for corpus work. When two linguistic phenomena are compared on the basis on the number of results returned by Google, the counts cannot be trusted. For instance, Véronis[2] analyzes problems with the logic of Google output and shows (among other things) that a search like (Chirac OR Sarkozy) produces *fewer* results than a search for a single term in the OR expression.

The problems with ordering the results and the amount of returned contexts have been addressed by several projects, such as KWiCFinder (Fletcher 2004) or WebCorp (Renouf 2003), which rely on AltaVista or Google queries, but present results in the form of traditional concordances. However, this does not solve

[2] `http://aixtal.blogspot.com/2005/02/web-googles-missing-pages-mystery.html`

the problems with counts, query language and richer linguistic information.

The ideal solution for corpus linguists would be a Google-like engine adapted to linguistic criteria. Kilgarriff (2001) discussed this idea under the name of D3CI (Distributed Data Distributed Collection Initiative), which would crawl the Web, collect a list of URLs to create a virtual corpus, which should be distributed over many servers. If a page from the list is not available at the time of querying, it can be replaced by any other page with similar characteristics (following the same methods as used by Google in their "Show similar pages" link). Unfortunately this approach has not been put into practice, probably because of the inherent difficulties involved in maintaining and querying a distributed corpus. Later on, the same idea was used by Oxford University Press (Kilgarriff, personal communication) for development of a new Internet-based representative corpus for English that should replace the BNC in dictionary development within OUP. However, the results of this project are not available for the academic world and are restricted to English only. Similarly, the WaCky initiative recently started crawling the Web to collect large corpora for English, German and Italian (see Bernardini et al. this volume).

A simpler methodology that does not involve crawling can be based on collecting a list of URLs from the Internet using the existing crawl index of search engines. For instance, Phil Resnik and his colleagues (Resnik and Smith 2003) extended their technique for developing parallel corpora to collect a list of URLs of Russian pages from the Web archiving engine `http://www.archive.org`. However, their list contains links to many pages that no longer exist or to pages that do not contain instances of connected text, such as price lists, collections of photos, etc. The same problem of retrieving pages with connected text appeared in a study by Fletcher (2004), who collected an Internet corpus by making a series of queries for the ten highest frequency words in the BNC,

retrieved a corpus of about 7,000 documents (after filtering duplicates) and reviewed all of them manually. In the result he selected 5,000 documents with a reasonable amount of connected text (i.e., he discarded about 30% of documents) following the estimation of Ide et al. (2002) for the minimum of 2000 words as an indicator of connected text. A similar technique was also used in CorpusBuilder (Ghani et al. 2003), though they did not evaluate the composition of their results and even give very little information about the size of their corpora. Baroni and Bernardini (2004) developed BootCaT, a tool for downloading webpages through the Google API and applied it to creating specialized corpora. Further, Ueyama and Baroni (2005) used the tool for creating a general-purpose Japanese Web corpus of approximately 3.5 million words using query words taken from an elementary Japanese language textbook. However, these experiments were not aimed at using Internet for building a BNC-like corpus, i.e., a corpus of at least 100 million words covering a variety of text types and domains.

The aim of this paper is twofold. First, I investigate the possibility to develop a BNC-like corpus for a number of different languages (Chinese, English, German, Romanian, Ukrainian and Russian). Second, I present an evaluation of the collected corpora using their composition and frequency lists for some of the languages (English, German and Russian). Since large balanced corpora are available for English and Russian, we can compare our Internet corpus against their content. For English we use the BNC, for Russian – the Russian Reference Corpus (RRC). Its pilot version used in this study contains about 35 million words, 45% of which is fiction, the rest is split between newspapers and various domains; for more information cf. Sharoff (2004).

2 DIY manual for a BNC

The method for collection of a large corpus for language X is based on BootCaT (Baroni and Bernardini 2004) and comprises four basic steps:

1. word selection: choose 500 word forms that are frequent in language X;
2. query generation: produce 5,000-8,000 queries, each of which contains 4 words from the word list from Step 1
3. downloading: send the queries to a search engine and collect the top 10 URLs returned for each query
4. post-processing: solve problems with encoding, boilerplate, duplicates

Now we will explain the rationale for the parameters used in each step.

2.1 Step 1: Word selection

Words in the query list should be sufficiently general, i.e., they should not indicate a specific topic. If a word like *Zeppelin* was used in the query list, this would create a bias in our corpus towards texts from the history of aviation or hard rock. On the other hand, function words frequently occur in pages that do not contain complete sentences, such as catalogues, captions for photos, price lists. For instance, *from* can bring a page from a holiday catalogue with a photo and caption: *Two weeks in Toscana, prices from 300 £*. If the goal of corpus collection is to provide examples of language use in connected texts, such pages should be avoided.

Many common frequent words indicate a particular topic, such as *work* or *room*. However, they can be used in the word list, as they do not bias the corpus because of their polysemy. Even

when they are not polysemous, common words can still be used in frequency lists, if they indicate a large number of situations (see examples with *work* and *room* in section 2.2 below).

Some studies, e.g., Kilgarriff and Grefenstette (2003); Ghani et al. (2003), considered the need to select words that are unique to the language of corpus collection. For instance, according to such views it is not advisable to use *restaurant*, as this word exists in several different languages. However, the query stage in the proposed methodology uses the language filter of a search engine, which by itself rarely makes mistakes in page classification. What is more, the presence in each query of three other frequent words in the target language should eliminate pages in "wrong" languages.

Since general search engines (such as Google or Yahoo!) do not perform lemmatization, we have to rely on lists of word forms only. This can in principle distort results in the case of languages with elaborate morphology, such as Arabic, Romanian or Russian, in which a word may have 10-20 forms or more. Thus, a query based on exact word forms in such languages operates with words that are much rarer in comparison to English. For instance, two lemmas *high* and высокий are good translation equivalents having roughly the same rank and frequency in English and Russian, as their position in the respective frequency lists is around 180 and the frequency is around 500 instances per million words (ipm). However, the frequencies of the exact forms *high* and высокий are quite different: 290 ipm for *high* with the rank of 264 vs. 34 ipm for высокий with the rank of 2,140 (the shift of its rank also reflects the number of forms of more frequent words). This means that for languages with rich morphology in the end we will find fewer pages, because in those languages we use less frequent word forms. Fortunately, this did not cause problems for our study, because many webpages exist in the languages under study (Romanian, Russian, Ukrainian) anyway, so we can find a sufficient number of hits for each query. At the same time, in languages with richer

morphology it is possible to use only forms that are more likely to appear in connected text, such as verbs, because the presence of a verb indicates that there is a clause.

For English and Russian we used 500 frequent common words from the frequency lists from respective representative corpora. For English we used the frequency list of word forms collected by Adam Kilgarriff from the BNC. For Russian we used the frequency list of the RRC. For Chinese, German and Romanian we also started with frequency lists from existing corpora, which exhibited some bias towards news items. For Chinese it was the "Gigaword" corpus, consisting of Xinhua newswires (thus excluding the Taiwanese section of the "Gigaword" corpus, because it uses another version of Chinese characters). For German, the frequency list was based on the list of word forms from the IDS corpus from Institut für Deutsche Sprache. Even though the IDS corpus contains a variety of text types (including some fiction and texts from science and humanities), it is biased towards news sources. This is reflected in its frequency list: the word *SPD* (the name of a German political party) is more frequent in it than *ja* "yes", *Kinder* "children" or *Frau* "woman". We extracted from it the list of the most frequent 500 words which start with lower-case letters (adjectives, adverbs and verbs) and are not specific with respect to a topic, e.g. *häufiger* "more frequent", *wünscht* "wants", etc.

If we want to develop a corpus for a language and we do not have access to a frequency list, we can rely on intuition in creating the word list for queries, because the exact frequency of words is less important than selection of common frequent words that do not point to a specific domain (this was the case with the Internet corpus for Ukrainian).

We can use more words from the frequency list than the original suggestion of 500. However, this increases efforts put into development of the query list (we spend more time cleaning the list from words we do not want) and increases the number of topic-

specific words as we progress along the frequency list to less and less frequent words.

2.2 Step 2: Query generation

We use four common words in a query following the requirement to get pages that contain relatively long pieces of connected text, with a smaller number of *noisy pages* in the form of price lists, tables, lists of links, etc. Shorter queries and the use of function words result in more noise. Function words are invariably used in broken sentences, such as catalogues or lists of headlines, which are not ideal candidates for a corpus. The presence of one-two common words also does not guarantee an instance of connected text. For instance, the first page returned by Google for the query *work* AND *room* includes several links to pages which do not contain stretches of connected text, such as `http://www.readingroom.com/aboutus/featuredwork.cfm`.

At the same time, a four-word query is much more likely to yield a page with narrative prose. For instance, the top ten pages produced by the query *work room hand possible* all have stretches of narrative prose ranging from two to five thousand words (not counting navigation frames). The pages retrieved also refer to a variety of domains, including a selection of summaries from Yahoo! news, pages on political debates, orthopedic surgery, forensic investigation analysis, classes offered in an art center, a blog on maps, descriptions of furniture, electronic tools, fiction books and historical events. Even more specific words, such as *Scottish* in the context of a four-word query bring a variety of topics. For instance, the query *deep houses resources Scottish* returns pages devoted to history, architecture, politics, technology (production of energy), funding guides, etc.

However, if we use queries longer than four words, the number of pages returned gets smaller, so that the result will not qualify as a random snapshot of the Internet. Even for English (the language

most widely used on the Internet) a query of eight words frequently produces few hits or the result consists of duplicate pages. It is possible to relax the condition for four words in a query for languages which do not have sufficient number of Internet pages. For instance, we used queries of three words for collecting the Romanian corpus. Even though there is sufficient amount of pages in Romanian, our task was to collect a corpus with proper encodings of diacritics, which are frequently omitted in Romanian Internet pages.

BootCaT has a mechanism for automatic generation of a random list of N-tuples out of the original word list. In this experiment it has been extended with the mechanism of prefixing random stings with a specific string to achieve the following functionality. Search engines can restrict the search to a variety of languages using their own linguistic filters. However, if the language for which we want to collect a corpus is not covered, each query can be complemented with a couple of very frequent function words that are not used in cognate languages, e.g. for detecting Ukrainian we prefixed має OR її ("has OR her") to each query.

2.3 Step 3: Downloading

In the reported experiment we used the Google API (application programming interface) via BootCaT. Since then another API for Yahoo! has been made available. For each query we take 10 top URLs returned by the Google API and use them for further processing. In the current setup we used 5,000 queries, which resulted in 50,000 URLs. However, some URLs can be found more than once as a result of different queries. The downloading step reduces the number of URLs further, because of the dynamic nature of the Internet: not all pages indexed by Google are available at the time of downloading. This may require additional queries to extend the database of URLs to reach the target corpus size, say a corpus of more than 100 million words requires about 35-40,000

pages, given that downloaded pages contain on average about 3-4,000 words. The list of successfully downloaded URLs is stored in the corpus database and can be used to recreate the corpus by other researchers.

The procedure can be repeated to enlarge the corpus up to the limit of all texts in this language indexed by the search engine. However, a corpus of 100 million words gives abundant lexicographic data for words common in the general language. According to our experiments with the languages under study, the top 25,000 words have at least 100 occurrences (words at the end of the 25,000 word list in English include *exploitative, lithograph, neutrophil,* and some proper names). A concordance of 100 lines provides sufficient evidence for lexicographers, especially given that such words are typically monosemous, cf. the experience in development of the COBUILD dictionary (Sinclair 1987). Words that do not provide this evidence in a 100 million word corpus (such as those with 10 occurrences or less) are rare or misspelled words e.g. *oystercatcher* or *somtimes*. A study of terminology in the field of oystercatchers (a bird of the family of Haematopodidae) will require a specialized corpus.

The upper limit for an Internet corpus depends on what is a reasonable size for its storage and reasonable time for producing concordances. Currently the Corpus WorkBench, the tool we use for indexing and querying it, limits the size of annotated corpora (with POS and lemma tags) to about 200 million words. Some studies, e.g. Kilgarriff and Grefenstette (2003), show that many unsupervised algorithms (such as those for word sense disambiguation) steadily improve their performance on larger corpora reaching the size of one billion words. So for some applications it might be advisable to collect a larger corpus.

2.4 Step 4: Post-processing

Pages collected in the previous step are subjected to postprocessing. First, it is necessary to unify the page encoding, which is also not always specified in the page attributes (Russian pages can come in 6 different encodings for Cyrillic characters). Second, we use the `lynx` browser to convert pages from HTML into plain text. This works better than frequently used ad-hoc Perl filters, as it removes HTML add-ons, including javascripts or comments, but does not lose information on character encodings (`lynx` has options `display_charset` and `assume_local_charset` to render them correctly once we identified them for every page). Another advantage of `lynx` is that after removing HTML tags it leaves traces of links in the original document, so that we can use simple heuristics to remove navigation frames (such as the density of links, which tend to appear mostly in navigation frames). Finally we can filter out pages that are either completely identical (e.g. two copies of the GNU Public License) or almost identical (e.g. a page with navigation and its printer-friendly version). The simple procedure used for the Internet corpora reported on in the paper involved detection of exact duplicates only. Since then, Baroni and Zanchetta produced a tool for detection of shared n-grams in large text collections,[3] which helps in finding near duplicates using the shingling algorithm (Broder et al. 1997): if several identical n-grams appear in two documents, this is an indication that the two documents share significant part of their text.

This sequence of steps results in a clean corpus in plain text format using a single chosen encoding. Finally, in order to create a proper corpus out of this collection of plain texts, we need language-dependent morphosyntactic processing, such as tokenization (more important for Chinese and other languages without

[3] `http://sslmitdev-online.sslmit.unibo.it/wac/post_processing.php`

	I-EN	I-DE	I-RU
Number of tokens	126,643,151	126,117,984	156,534,391
Number of word forms	2,003,056	3,384,491	2,036,503
Number of lemmas	1,608,425	3,081,197	791,311
Number of URLs	42,133	31,195	33,811
Average document length (in words)	3,006	4,043	4,630

Table 1. Some statistics for Internet corpora

explicit word boundaries), lemmatization (especially for morphologically rich languages), as well as POS tagging.

A summary of the characteristics of the Internet corpora collected for English, German and Russian is given in table 1 (abbreviated as I-EN, I-DE and I-RU respectively). The size of the corpora varies slightly: the longest pages have been retrieved for Russian, so the Russian corpus is slightly bigger. The most significant difference is in the number of lemmas in the lexicon: 791,311 in I-RU vs. 3,384,491 in I-DE. This depends partly on the features of particular languages and partly on properties of tokenizers and lemmatizers. For instance, in German there are many compound nouns, which in other languages are typically decomposed into several words, e.g. *Fachhochschulratspräsident* (the president of the council of polytechnic universities). This increases the amount of separate forms and lemmas. The smaller number of lemmas in Russian can be partly explained by the larger number of word forms per lemma, as well as by more aggressive splitting done by the Russian lemmatizer used in the experiment (`mystem`),[4] which treats hyphens as word separators. In contrast, our English and German lemmatizers (respective versions of the TreeTagger[5]) treat the hyphen as a word character.

[4] `http://corpora.narod.ru/mystem/mystem.html`

[5] `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/`

3 What is under the hood?

In this section we use two methods to compare Internet corpora against standard manually-collected corpora such as the BNC, Reuters or Gigaword. The first method involves assessment of corpus composition using a text typology, which is similar enough to the one used in the BNC to allow comparison between the BNC and Internet corpora. The second methodology involves comparison of lists of the most frequent words taken from various corpora to show the most significant differences in their lexicon.

3.1 Composition assessment

The reported procedure produces a corpus of about 40,000 texts, which is not practical to assess in its entirety, so we have to choose a representative sample. The issue of the representativeness of a text collection in terms of the number of documents is frequently neglected in corpus studies, whereas statistics offers a straightforward procedure to estimate the symmetric confidence interval, which is frequently used for determining the size of a sample required in sociological studies or polls:

$$\sigma = \pm c\sqrt{\frac{p(1-p)}{N}}$$ (Upton and Cook 2001, 301)

where c is the percentage point (or the critical value) from the standard normal distribution appropriate to attain the desired confidence level, p is the estimated probability of an event, and N is the population sample required for the result to be within the given confidence interval with the given confidence. Note that the value of the interval does not depend on the size of the population. The only assumption is that the total population is significantly larger than the size of the sample. The confidence level refers to the probability that the real distribution measured on the complete population will be indeed within the symmetric interval. For

the same sample size N we can make a statement with confidence of 90% ($c = 1.645$) or 95% ($c = 1.96$), giving a slightly larger symmetric confidence interval in the second case.

The total *population* in the case of a sociological study refers to the total number of people or cases which constitute the subject of the study, such as the number of voters in a country, while the *sample* refers to the focus group the study is based on. In the case of corpus studies, an Internet corpus is itself a sample of the population, i.e., the content of the Internet for a particular language, which in its turn is a sample of the total language used in the society. However, in terms of statistical analysis of its composition, an Internet corpus of 40,000 documents represents the total population, from which we take a sample in the form of a subset of URLs.

Application of the above formula is based on two assumptions: the normality of the sample distribution and the approximation of the probability of an option. The first assumption is justified by random sampling from a much larger list. The second assumption involves replacement of the unknown value of p, the probability of an option, e.g. the proportion of texts written by men, with its estimation from the number of options in the respective category. Categories in the text typology described below have 3-8 options, so we can estimate p as $0.125 \leq p \leq 0.33$. Of course, we cannot always make the assumption that all options in a category have equal probability. However, the value of $p(1-p)$ does not vary much: for any $0 \leq p \leq 1$ it is always true that $p(1-p) \leq 0.25$ and it gets smaller for smaller values of p providing a more precise symmetric interval.

In short, this means that if we take a random sample of 200 documents from a text collection, we can achieve the confidence interval of $\sigma = \pm 5\%$ and confidence level 90%. A better approximation of the corpus composition within the interval of $\pm 1\%$ with 95% confidence will require a much larger sample, of about 1,500

documents. In our experiments we used samples consisting of 200 documents, so the figures reported below assume the confidence interval of $\sigma = \pm 5\%$ with confidence level 90%.

3.1.1 Text typology and detection criteria

Assessment of the corpus composition requires a text typology to annotate texts in the sample. Existing research in corpus studies has produced two theoretically sound text typologies. First, an extensive text typology has been developed for coding texts in the BNC, but it paid more attention to the bibliographic classification of corpus files and did not touch some issues concerning the function a text carries in the linguistic community. Second, the European Advisory Group on Language Engineering Standards (EAGLES) produced text typology guidelines in work headed by John Sinclair (EAGLES 1996; Sinclair 2003). The EAGLES guidelines include functional categories, however, they do not cover many text types that are frequent in general-purpose corpora or webpages, such as types of newspaper texts or advertisements. Finally, the text typologies from the BNC and EAGLES offer too many options in the sense that if we use all the categories available for coding even a sample of a corpus, the coding will take a lot of time and the results will be less reliable.

We attempted to develop a small set of categories and rules for assigning values to those categories. This set of proposed categories is specific enough to describe the great majority of Internet pages with adequate sociolinguistic precision, but at the same it is quite small, so that each document requires no more than 5–8 choices from the list of categories. The coding itself was done using the `Systemic Coder` (O'Donnell 1995), which provides an interface for prompting choices for each text and allows basic statistical analysis of the results.

Another requirement for the set of categories is the reliability of information provided in Internet pages for detecting their values.

For instance, the gender of the author can be reliably identified in the languages used in the study by his/her first name, if it is given, e.g. *John* vs. *Mary*. There are relatively few cases when this cannot be done, either because it is ambiguous, like *Chris* in English, or the sex association is not known to the coder, as is the case with *Cody*. The sex of an unknown author sometimes can be guessed from semantic clues, e.g. if the author refers to *my husband*, or from grammatical properties, such as gender agreement in Russian (я была... – I was-fem). At the same time, a guess about the age of the author or the size of the intended audience is much less reliable, so these were not included in the classification scheme.

We assess each text using 5 categories: authorship, mode (aka channel), knowledge expected from the audience, the aim of text production and the generalized domain. The basic set and the order of categories follows the EAGLES guidelines and corresponds to the degree of certainty in coding values of those categories: it is quite easy to code the authorship, while many texts cover several domains at once, so the choice of the domain is less reliable. In order to reduce possible ambiguity in choosing the values of categories we provide explicit instructions for filling their values on the basis of observable features of texts. In a trial study four colleagues were asked to code a sample of 100 texts according to the proposed typology. They all completed the task in less than an hour with very small variation in the set of assigned categories.

Full results of assessment of the composition of automatically acquired corpora are shown in table 2.[6] The English and Russian Internet corpora can also be compared against data obtained from representative corpora for those languages, though the comparison cannot be complete, as neither the BNC nor the RRC classify pages with respect to the purpose of their production. The *audience level* code from the BNC cannot be directly compared against

[6]Since additional annotators did not assess the complete sample, the results listed in the table are based on my own counts.

the knowledge expected from the audience according to our typology, while in the RRC there is no coding for this category at all.

In the following subsections we describe the set of categories in detail and give instructions for making decisions about choosing their values.

3.1.2 Authorship

Information about the authorship uses the following values:

- **single** – created by a single named author. We also classify the sex of explicitly named single authors, in so far as this can be detected using the name and other lexical or syntactic clues (such as references to author's husband, third person pronouns referring to the author, grammatical agreement, etc).
- **multiple** – created by several named co-authors.
- **corporate** – created by a corporate author (in this case there is a corporate copyright statement and a human author is not given; this applies also to texts created by governments and non-profit organizations). There can be some inconsistency here: a newsitem in the newspaper can lack the name of its author, while a feature article, which still carries a corporate copyright statement, can have an explicit author's name. In the latter case, the decision should be made for the single named author. On the other hand, a letter for investors has been claimed to be written by the CEO of a company, but since it represents the position of the company and most probably it was edited by the whole board of directors (if not external consultants), it should be coded as corporate. The same applies to such documents as Papal Encyclicas or declarations in the name of the heads of governments.

- **unknown** – no information about the author is available on the page nor can it be inferred without significant extra efforts.

The results reported in table 2 show that Internet corpora in comparison to traditional representative corpora, contain significantly more texts coming from corporate sources (44% for I-EN vs. 18% for the BNC), while they consistently underrepresent female writers (23% of texts in I-EN are written by men vs. just 3% by women in comparison to the 28% vs. 13% split in favour of male writers in the BNC).

3.1.3 Mode

The classification of texts with respect to their mode follows the EAGLES guidelines using the following values:

- **written** – traditional written texts, including online newspapers, homepages, etc;
- **spoken** – transcripts of sound-wave recordings, including interviews;
- **electronic** – spontaneous communication, such as emails, electronic forums or chat rooms.

The EAGLES guidelines introduced the electronic mode "to emphasize that language transmitted in electronic media is not quite the same as the older established modes". For the purposes of coding webpages (all of which exist in electronic form), the use of the electronic mode was restricted to spontaneous electronic communication. The separation is important, because in comparison to traditional written texts they are similar to spoken communication in the spontaneity of their production (like face-to-face or telephone conversations). However, they are *not* spoken texts, so

		BNC	I-EN	RRC	I-RU	I-DE
Authorship	Corporate	18%	44%	-	38%	51%
	Male	28%	23%	50%	18%	13%
	Female	13%	3%	25%	6%	2%
	Unknown	4%	11%	16%	15%	14%
	Multiple	36%	19%	9%	23%	20%
Mode	Written	90%	86%	100%	84%	90%
	Electronic	0%	13%	0%	16%	9%
	Spoken	10%	1%	0%	0%	1%
Audience	General	27%	33%	-	40%	61%
	Informed	47%	45%	-	46%	31%
	Professional	26%	22%	-	14%	8%
Aim	Discussion	-	45%	-	47%	45%
	Information	-	11%	-	4%	25%
	Recommendation	-	34%	-	35%	21%
	Instruction	-	6%	-	3%	5%
	Recreation	-	4%	-	11%	4%
Domain	Life	27%	14%	51%	25%	12%
	Politics	19%	12%	18%	10%	21%
	Business	8%	13%	3%	7%	5%
	Natsci	4%	3%	2%	3%	1%
	Appsci	7%	29%	3%	19%	18%
	Socsci	17%	16%	16%	5%	8%
	Arts	7%	2%	6%	2%	4%
	Leisure	11%	11%	1%	26%	31%

Table 2. Comparison of corpus composition

they lack prosodic information, which is compensated by capitalization or new means of expression, such as emoticons and smileys. Electronic texts also often exhibit a large number of typos and non-standard choices.

Only 10% of the BNC consists of spoken texts, because collection of a larger spoken corpus was not considered to be practical. In Internet corpora we find very few instances of transcripts of spoken language, but spontaneous language is predominantly represented by discussion forums, so electronic texts correspond to 16% of the Internet corpus for Russian, 13% for English and 9% for German.

3.1.4 Audience

It is frequently impossible to make a reliable judgment with respect to values of the audience parameters using the full set of categories from the BNC and EAGLES text typologies. For instance, the BNC index uses identical codes for describing an article from *The British Journal of Social Work* (text GWJ) and an article on French smoking habits from the tabloid *Today* (CEK): both are published in periodicals and belong to the domain of humanities. The BNC typology provides a code distinguishing the audience level, but both texts are coded as medium.

In our experience the judgment on such audience parameters as its size or level are hard to make, but we can reliably code the level of *knowledge* expected from the audience to read a text:

- **general** – no knowledge about the topic is required for reading this text, e.g. a text on ulcers from the *BBC* website. Such texts are written for the broadest general public. They refrain from using terminology that the general public is not expected to know.

- **informed** – some general knowledge of the topic is required, e.g. a description of ulcers for medical students. Another example could be an explanation of the design of home theaters for audiophiles. Such texts are not very technical, but they do use a significant amount of specialist terminology.

- **professional** – significant prior knowledge about the domain is required for reading a text, e.g. an article in the *Journal of Gastroenterology and Hepatology*. Such texts are written for professionals using many abbreviations, dense terminology, etc. They also appear on specialized websites. This does not assume that the category is limited only to topics from respected professions. A discussion of the number of "ingots for GM tinkering" in *Ultima Online* is classified

as aimed at the professional audience as well.

The exact boundaries between texts aimed at the general, informed or professional audiences are vague, but in the vast majority of cases the decision is clear. The instruction for coders states

> If you can easily understand the text content, choose **general**; if you can in principle understand what the text is about, but it contains special terminology, choose **informed**; if you cannot understand the text, choose **professional** (if you are a specialist in the domain of the text, try to imagine yourself to be a layman)

In terms of their composition, Internet corpora contain a good balance of these three categories, with the prevalence of texts being aimed at informed audiences, e.g. 33% for general, 45% for informed, 22% for professional audiences in I-EN.

3.1.5 Aims of text production

This is the classification of texts according to their function in the society, as borrowed from Sinclair (2003), but with some modifications outlined below:

- **discussion** – texts aimed at discussing a state of affairs (e.g. articles in newspapers, academic papers, travel stories).
- **recommendation** – recommendations differ from discussions as they provide an incentive for doing or abstaining from doing something; examples of subclasses are: **advice, legal, advertisement**.
- **recreation** – the primary purpose of writing such a text is for leisure-time reading; the two important subclasses are **fiction** and **nonfiction**, further subclasses of fiction and

nonfiction can be distinguished, but they are too rare on the Internet to warrant this. This category is not necessarily concerned with leisure activities (cf. the subtypes of text domains discussed below).

- **instruction** – such texts are aimed at educating their readers; the following subclasses can be used: **manual** (e.g., recipes, flat-pack assembly or software manual pages; they typically come in the form of itemized lists), **practical-how-to** (this category encodes more descriptive text varieties in comparison to manuals, the most frequent example in this category among Internet texts is a FAQ), **textbook** (on the Internet we typically do not have complete textbooks, but explanations and introductory material on various topics, e.g. a Perl tutorial; this is the most discoursive type of instructive texts).

- **information** – texts whose primary purpose consists in providing information. Sinclair (2003) restricts the category to reference compendia, but in corpora we find many other cases, such as: **reference** (dictionaries, encyclopedias), **data** (police reports, summaries, minutes of project meetings, etc), **news-reports** (e.g. a message informing about an earthquake differs from a newspaper article about rescue efforts, the latter being classified as **discussion**). Note that this category is limited to texts only concerned with data dissemination. A discussion of the history of the Tory party in the *Wikipedia* is classified as **information**, while the Tory election manifesto is **recommendation**.

There are some borderline cases between **discussion** and **recommendation**, but in the majority of pages the distinction is clear: if it is evident that a text tries to persuade the reader to become a potential customer or supporter, it is classified as **rec-**

ommendation, a text without obvious propaganda is **discussion**. If the tests for other categories do not produce convincing results, the general rule for coding text production is to choose **discussion**.

A classification of this sort is used neither in the BNC nor in the RRC, so Internet corpora have no basis for comparison. However, the three Internet corpora being compared are quite similar with respect to aims of their production. Internet texts most typically discuss a topic or give recommendations (most typically by advertising products, services or political movements).

Texts aimed at **recreation** are treated as an important category in traditional corpora (fiction constitutes 17% of the BNC and 49% of the pilot version of the RRC, though the latter figure will be lower in the final version). However, because of copyright restrictions, published fiction texts are relatively rare on the Internet (especially in English and German, where they constitute just 3-4% of the Internet corpora). Texts aimed at recreation are more frequent in I-RU (11%), including OCR'd versions of fiction texts and exchanges of jokes, but still they are relatively rare.

3.1.6 Domain

The EAGLES guidelines mention the frequent variation of topics within a single document or conversation and reject the applicability of any general classification system (such as Dewey Decimal Classification). Instead, they list domains considered in various studies of terminology and corpora and refer to the unsuitability of "trying to arrange a hierarchy of simple topic labels". However, in practical terms the offered list of some 30 domains is too fine-grained. What is more, a webpage can be the subject of a far more delicate classification, which, nevertheless, should start from a node in the hierarchy.

Even though any classification of topics is not complete, we propose to use eight general categories for classifying webpages.

- **natsci** (maths, biology, physics, chemistry, geo, ...)
- **appsci** (medicine, computing, ecology, engineering, military, transport, ...)
- **socsci** (law, history, philosophy, psychology, sociology, language, education, ...)
- **politics**
- **business**
- **life** (a general topic that is used for fiction, conversation, etc.)
- **arts** (visual arts, literature, architecture, performing arts)
- **leisure** (sports, travel, entertainment, fashion...)

The labels associated with categories whenever possible follow the practice of the domain codes used in the BNC, but some have been changed to reflect additional dimensions of classification, e.g. **life** incorporates fiction (imaginative texts in the BNC), as well as weblogs on dating or parenting of a child; **world affairs** from the BNC is treated as **politics**. In parentheses we list examples of subclasses of the respective categories, which do not constitute a closed-class list, but can help in making the decision for classification of a page. Basic categories on the other hand do constitute a closed-class list to choose from.

There are fewer texts from arts, humanities and social sciences in Internet corpora in comparison to their traditional counterparts, e.g. 16% for **socsci** in the RRC vs. 5% in I-RU. Even though the figures for English look closer (17% in the BNC vs. 16% in I-EN), the vast majority of texts considered as **socsci** in the English Internet corpus are legal texts (legislation, law reports, terms and conditions, etc), not texts in history, linguistics or education as

in the BNC. At the same time there are many more texts from technical fields (**appsci**) on the Internet: 7% in the BNC vs. 29% in I-EN (Internet texts most frequently belong to such subdomains as computer science, medicine or construction industry).

If we compare this data against the Reuters corpus (a newswire corpus annotated with domain codes), we will find that 56% of the Reuters corpus consists of financial news (its C, E and M subcategories), contrasting with 13% of business texts in the Internet corpus (8% in the BNC). At the same time less than 0.5% of texts in the Reuters corpus is classified as science (GSCI), which includes the **natsci**, **appsci** and **socsci** categories taken together. What is more, texts in the Reuters corpus are obviously not aimed at discussing scientific topics or teaching about them, but mostly aimed at giving information in the form of news reports. This suggests again that Internet corpora can be claimed to be more representative than newswire corpora such as Reuters or Gigaword.

3.2 Comparison of word lists

Assessment of the corpus composition involves a significant amount of manual coding and implies near-native knowledge of the language and culture for which the corpus has been created. The comparison of frequency lists is a much faster way of understanding the major differences between the newly acquired corpus and a known benchmark corpus and judging how significant they are. Also unlike the corpus composition exercise, which starts with a predefined set of categories, comparison of frequency list is driven exclusively by data found in corpora (even though it is influenced by the results of tokenization and lemmatization).

Among various methods for comparing frequency lists we choose the log-likelihood statistic, since this has been suggested to provide the most reliable method for comparing frequency lists (Rayson and Garside 2000).

The computation of the log-likelihood statistic is based on the following contingency table:

	Corpus 1	Corpus 2	Total
Frequency of word	a	b	a+b
Frequency of other words	c-a	d-b	c+d-a-b
Corpus size	c	d	c+d

Then the expected values E1 and E2 and the log-likelihood value G2 are calculated as:

$$G2 = 2(a \ln(\frac{a}{E1}) + b \ln(\frac{b}{E2})); E1 = c\frac{a+b}{c+d}; E2 = d\frac{a+b}{c+d}$$

In the study reported below we calculated log-likelihood values for the frequency of lemmas or word forms in two corpora, took words with the highest values and listed separately words that are more frequent (overused) and less frequent (underused) in the second corpus in comparison to the first. The analysis should highlight statistically significant differences between the frequency lists and can suggest ways in which one corpus is less balanced than the other. For the sake of space, the tables show only the 10-12 words with the most significant log-likelihood scores, but in examples we occasionally discuss some other words with high scores.

First we take two corpora with known composition and compare the frequency list of a newswire corpus (Reuters) against a representative corpus of general language (BNC). In this step we identify the differences between the lexicon of a representative corpus vs. the lexicon of a newswire corpus (table 3).

Second, we compare an Internet corpus against a newswire corpus with known composition (the English Internet corpus against Reuters). In this step we also compare the German Internet corpus against the IDS corpus, the composition of which is unknown, but it is likely that IDS exhibits some features of a newswire corpus (because of relatively high frequency of hits from newspapers in

More in BNC	LL-score	More in Reuters	LL-score
you	6,005.14	say	8,559.54
I	5,271.42	percent	4,513.35
she	3,334.57	million	2,364.29
be	2,411.89	market	1,982.47
do	1,610.71	billion	1,518.25
they	1,502.79	bank	1,468.84
your	1,282.15	company	1,258.34
can	1,191.74	newsroom	1,240.37
what	1,090.53	share	1,214.84
my	1,023.56	tuesday	1,199.25

Table 3. BNC vs. Reuters

More in I-EN	LL-score	More in Reuters	LL-score
you	4,343.16	say	12,154.94
I	2,797.67	percent	3,424.40
your	2,731.17	million	2,103.23
or	1,845.60	market	1,943.17
my	1,262.80	bank	1,574.68
can	965.08	billion	1,270.30
this	899.29	newsroom	1,254.03
use	729.11	share	1,193.56
me	719.46	its	1,175.01
do	687.78	company	1,125.64

Table 4. I-EN vs. Reuters

More frequent in I-DE			More frequent in IDS		
Word form	**Gloss**	**LLscore**	**Word form**	**Gloss**	**LLscore**
ich	I	1,227.77	Mark	Mark	858.82
dass	that (new)	691.60	Uhr	hour	528.01
mir	me_{dat}	350.78	Prozent	percent	329.20
du	you_{fam}	376.29	daß	that (old)	307.32
mich	me_{accus}	273.24	sei	be-subjunc	291.95
the	-	266.27	dpa	dap	262.05
Ich	I	250.70	bis	to-temporal	258.87
Du	You_{fam}	241.12	Millionen	millions	235.37
of	-	198.39	gestern	yesterday	225.47
Beiträge	messages	178.55	SPD	SPD	181.97
Beitrag	message	155.29	sagt	said	177.19

Table 5. Comparing I-DE vs. IDS corpus

More in BNC		More in I-EN	
was	1,251.29	your	303.43
had	953.62	Posted	278.37
he	928.66	Web	262.23
she	912.82	program	255.15
er	909.30	Internet	228.45
her	795.37	site	217.36
Yeah	623.65	Click	201.91
it	580.80	Center	192.76
erm	578.10	online	189.36
his	496.03	Bush	177.53
I	415.54	email	177.42
said	398.64	information	174.04
Oh	385.29	New	168.38

Table 6. Comparing the BNC to I-EN

concordance lines). In doing this comparison we will try to show that Internet corpora differ from newswire corpora in more or less the same way as the BNC differs from the Reuters corpus (tables 4 and 5).

In the third step, we compare two representative corpora with known composition (BNC and RRC for English and Russian) against their Internet counterparts to study the differences between language use on the Internet and in general-purpose corpora. Word forms with the highest log-likelihood scores are shown in table 6. Word forms were used instead of lemmas because of differences in the lemmatization procedures used to produce frequency lists for the two reference corpora and automatically acquired Internet corpora. This boosts differences in lemma lists significantly without any underlying linguistic reason.

Tables 3 and 4 show that newswire corpora in comparison to both the Internet and the BNC overuse words referring to financial data (*million, Mark*), specific entities and institutions (*market, dpa*), other financial terms (*share,* also *analyst, trader, price*) and exhibit greater use of temporal markers that specify the date and time of an event (*Tuesday, Uhr*). Another specific feature of newswires is much greater use of reported speech, which is reflected in the overuse of such words as *say, sagen*. In German *sei/seien* (the subjunctive forms of *sein,* "to be") are also markers of reported speech, in particular, they are frequently used as copular verbs in this context, for example:

> *Jacques Delors pflegte zu sagen dass der Markt kurzsichtig sei und es deshalb politisch notwendig sei die Unterschiede zu verringern.*
> "Jacques Delors was accustomed to saying that the market was short-sighted and hence it was politically necessary to reduce the disparities."

At the same time words that are *less* frequently used in news-

wire corpora follow the same pattern as established by the comparison between the Reuters corpus and the BNC. Newswire corpora in comparison to the BNC and Internet corpora use fewer first and second person pronouns, question words (*what, welche*), modals (*can, muss*), mundane verbs (*go, gehen*). This means that the composition of automatically acquired Internet corpora reflects general language in a way similar to a manually constructed representative corpus.

Finally, table 6 shows the most significant differences between the frequency lists of word forms in representative corpora vs. Internet corpora. In addition to the above-mentioned technical reason (differences in lemmatization) the use of lists of word forms helps as it reveals more facts concerning the use of specific forms, such as *Posted* (capitalized and in the past tense), which is an indicator of the time when a message appeared on the Internet. The list of word forms also makes it clear that the BNC shows much greater use of past forms (*was, had, said*) and third person pronouns (*she, he, her,* it). This correlates with another study of the language used on the Web made by Fletcher (2004), who also remarks that

> the BNC data show a distinct tendency toward third person, past tense and narrative style, while the Web corpus prefers first and second person, present and future tense and interactive style.[7]

Words that are more frequent in the BNC include several interjections (*er, Yeah, Oh*), which frequently occur in transcripts in the spoken component of the BNC, as well as in fiction stories, as their authors use them to imitate spoken language. As discussed earlier, fiction is underrepresented on the Internet, while the lan-

[7]There may be several reasons why the first person pronoun *I* is in the list of words more frequent in the BNC. One possibility is that many Internet writers use the lower case *i* in this function.

guage of chat rooms makes very little use of hesitation markers such as *er*.

It is not surprising that words more frequent in Internet corpora include Internet-specific words (*Web, site, email*) or words related to interaction with it (*Click, program, Reply*), as well as words referring to hot topics at the time of corpus collection (*Bush, Yushchenko*). At the same time the differences between word frequencies in the Internet and representative corpora are much less significant than those for corpora based on newswires.

4 Conclusions and further research

The proposed procedure described in section 2 is applicable to any language with more or less significant Internet presence. The procedure can produce a large corpus (100-200 million words) which, as shown in section 3, can be considered as comparable to large representative corpora in terms of its size and coverage of various domains. What is more, the corpus can be considered as "open-source", as it exists as a set of URLs accompanied by additional open-source software for downloading the set of HTML pages and post-processing them (i.e., removing navigation frames, tables, duplicate pages, etc). If the parameters of an Internet corpus are described with adequate precision, it can function as a benchmark used by other researchers in the same way as the BNC. For instance, everyone can use the BNC to compare the frequency of occurrences of *strong tea* and *powerful tea* and make conclusions about their most typical contexts, for instance, by referring to the fact that *powerful tea* occurs three times in a single text and the reason why it occurs there is that that text is exactly on the topic of corpus linguistics and collocations and the example is used to illustrate collocations impossible in English.

If we claim that an Internet corpus is useful as a benchmark for studying language X, it is necessary to understand how stable the

benchmark is. If you do your study for English on the basis of the BNC, there can be minor variations depending on the version of the BNC you are using. However, changes concern a tiny portion of the whole corpus: the number of occurrences of *powerful tea* will not change. Kilgarriff (2001) defended the possibility to use Internet corpora, which are dependent on the transient nature of the Internet, by referring to a scientific study of water taken from river Lune: you cannot expect that molecules are exactly the same, yet the study is replicable. However, chemical analysis provides ways for measuring how replicable the study is.

If we distribute an Internet corpus in the form of URL lists, one possible measure can concern the half-life of those lists, i.e., we can measure how many URLs from the original list are still accessible after a certain period and how much of the content of the respective pages is the same. Research in this area is still in its infancy, so we would like to study it more closely in collaboration with Marco Baroni. The parameters for studying the URL half-life will include the number of URLs retained, the proportion of the text remained exactly identical, differences in the frequency of retrieved words, differences between URL sets for languages.

In addition to studying changes in Internet corpora derived from a fixed set of URLs, one can study variations caused by differences in the collection procedure. Ueyama and Baroni (2005) conducted a study of two Japanese corpora collected according to the same procedure using the same list of query words, but the first corpus was collected in July 2004, the second one in April 2005. The study shows that the composition of the two corpora varies considerably (even the intersection between the two sets of URLs is below 20%). It would be interesting to extend this research by studying the rate of change of Web-derived corpora using the influence of several other parameters apart from the variation in time, such as the differences between:

- languages and cultures: all languages exhibit explosive grow

on the Internet, but one can expect that the rate of change for languages currently less present on the Internet is more significant;

- search engines: we can study the difference between corpora derived using Google, Yahoo! or our own crawling engines (also using different methods of crawling);
- sets of query words selected from various sources (such as frequency lists) according to the same procedure (such the one outlined in Step 1 below)
- procedures for selecting query words: we can also study the difference between corpora produced using function words, adjectives only, words specific to a domain (e.g. set of headwords from Encyclopedia Britannica), etc.

More in I-EN2	LL-score	More in I-EN1	LL-score
I	143.14	tea	70.47
June	120.60	Christmas	34.21
Posted	99.64	dog	27.17
book	62.09	and	24.01
Definitions	51.45	Tea	22.37
blog	50.74	Speaker	21.00
that	47.98	PST	20.34
think	47.02	Feb	20.21
References	45.66	dogs	19.46

Table 7. Comparing two Internet corpora collected using different query words as seeds

As for now we can briefly show preliminary results regarding the URL half-life and corpus variation. Two English Internet corpora were collected in February and June 2005 respectively, using two sets of 500 query words without any intersection between the two sets. Both lists were extracted from the BNC frequency

list. The June list included the most frequent words, e.g. *chance, minutes, simple, thank*, while the February list consisted of less frequent common words, e.g. *opinion, purpose, suddenly, unemployed*. An experiment in August, 2005 involved downloading a random selection of 1,000 URLs from each of them. 934 URLs from the February corpus and 982 URLs from the July corpus were still available. Further experiments are necessary for determining the rate of degradation. As for the difference caused by sets of query words, table 7 shows the comparison between the frequency lists of the February corpus (I-EN1) and the June one (I-EN2). The differences (measured by the log-likelihood score) are much less significant in comparison to those reported in tables 4 and 6.

An interface to Chinese, English, German and Russian corpora, respective URL lists, lists of queries and the results of corpus assessment are available from `http://corpus.leeds.ac.uk/internet.html`.

References

Aston, G. and Burnard, L. (1998). *The BNC handbook: Exploring the British National Corpus with SARA*, Edinburgh: Edinburgh University Press.

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. In *Proceedings of LREC 2004*, 1313-1316.

Broder, A., Glassman, S., Manasse, M. and Zweig, G. (1997). Syntactic clustering of the Web. *Proceedings of the Sixth International World-Wide Web Conference*.

Cieri, C. and Liberman, M. (2002). Language resources creation and distribution at the linguistic data consortium. *Proceedings*

of the Third Language Resources and Evaluation Conference (LREC02), 1327-1333.

EAGLES. (1996). Preliminary recommendations on text typology. Technical Report, EAGLES Document EAG-TCWG-TTYP/P, EAGLES.

Fletcher, B. (2004). Making the Web more useful as a source for linguistic corpora. In Connor, U. and Upton, T. (eds.) *Corpus linguistics in North America 2002*, Amsterdam: Rodopi.

Ghani, R., Jones, R.and Mladenić, D. (2003). Building minority language corpora by learning to generate Web search queries. *Knowledge and Information Systems* 7(1), 56-83.

Ide, N., Reppen, R. and Suderman, K. (2002). The American National Corpus: More than the Web can provide. *Proceedings of the Third Language Resources and Evaluation Conference (LREC02)*, 839-844.

Kilgarriff, A. (2001). The Web as corpus. *Proceedings of Corpus Linguistics 2001.*

Kilgarriff, A. and Grefenstette, G. (2003). Introduction to the special issue on the Web as corpus. *Computational Linguistics* 29(3), 333-347.

O'Donnell, M. (1995). From corpus to codings: Semi-automating the acquisition of linguistic features. *Proceedings of the AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation.*

Rayson, P. and Garside, R. (2000). Comparing corpora using frequency profiling. *Proceedings of the Comparing Corpora Workshop at ACL 2000*, 1-6.

Renouf, A. (2003). WebCorp: Providing a renewable data source for corpus linguists. *Language and Computers* 48(1), 39-58.

Resnik, P. and Smith, N. (2003). The Web as a parallel corpus. *Computational Linguistics* 29(3): 349-380.

Robb, T. (2003). Google as a quick 'n' dirty corpus tool. *Teaching English as a Second or Foreign Language* 7(2).

Rose, T. and Stevenson, M., and Whitehead, M. (2002). The Reuters corpus volume 1: From yesterday's news to tomorrow's language resources. *Proceedings of the Third Conference on Language Resources and Evaluation (LREC02).*

Sharoff, S. (2004). Methods and tools for development of the Russian Reference Corpus. In Archer, D., Wilson, A. and Rayson, P. (eds.) *Corpus linguistics around the world*, Amsterdam: Rodopi.

Sinclair, J. (2003). Corpora for lexicography. In van Sterkenberg, P. (ed.) *A practical guide to lexicography*, Amsterdam: Benjamins, 167-178.

Sinclair, J. (ed.) (1987). *Looking up: An account of the COBUILD project in lexical computing*, London: Collins.

Ueyama, M. and Baroni, M. (2005). Automated construction and evaluation of a Japanese Web-based reference corpus. *Proceedings of Corpus Linguistics 2005.*

Upton, G. and Cook, I. (2001). *Introducing statistics*, second edition, Oxford: OUP.

Volk, M. (2002). Using the Web as a corpus for linguistic research. In Pajusalu R. and Hennoste T. (eds.) *Tähendusepüüdja. Catcher of the meaning. A Festschrift for Professor Haldur Õim*, Tartu: University of Tartu.

Evaluation of Japanese Web-based Reference Corpora: Effects of Seed Selection and Time Interval

Motoko Ueyama

1 Introduction

The World Wide Web is an enormous resource of accessible textual documents, and there is by now a considerable amount of work on using the Web as a source of linguistic data for a variety of linguistic and language technology tasks (see, e.g., the papers collected in Kilgarriff and Grefenstette 2003). A promising approach to the use of the Web for linguistic research is to build corpora by running automated queries to search engines, retrieving and post-processing the pages found in this way (e.g., Ghani et al. 2003; Baroni and Bernardini 2004; Sharoff this volume). This approach differs from the traditional method of corpus construction, where one needs to spend considerable time finding and selecting the texts to be included, but can have perfect control over contents. With the aforementioned automated methods, the situation is reversed: one can build a corpus in very little time, but without good control over what kinds of texts are included in the corpus. These automated methods, despite the almost complete absence of quality control, have made it possible to construct written corpora for linguistic research in a quick and economic manner. This is good news for researchers who urgently need large-scale balanced corpora (i.e., something equivalent to the British National Corpus) for the language of their interest, but who have no access to such

corpora. This is the case for researchers working on the majority of the world's languages, including Japanese (see Goto 2003 for a survey of Japanese corpora currently available for research purposes).

The pioneering work in the automatic construction of Web corpora has been done by the CorpusBuilder project (see, e.g., Ghani et al. 2003) that developed a number of related techniques to build corpora for languages with fewer NLP resources. Ghani and colleagues evaluated the relative performance of their proposed methods in terms of quantity of retrieved pages. However, they did not provide a qualitative assessment of their corpora, such as a classification of the pages. Baroni and Bernardini (2004) introduced the BootCaT tools, a free suite of Perl scripts for the automated, possibly iterative construction of corpora via Google queries. While the tools were originally intended for the development of specialized language corpora and terminology extraction, they can also be used to construct general-purpose corpora by selecting appropriate query terms. The BootCaT tools were used for this purpose by Baroni and Ueyama (2004), Ueyama and Baroni (2005), Sharoff (this volume).

As mentioned earlier, Japanese is one of the languages for which general balanced corpora are not available. In the aforementioned studies (Baroni and Ueyama 2004; Ueyama and Baroni 2005), we built two Japanese Web corpora with the BootCaT procedure. In this study, we build another Japanese Web corpus with the same procedure, and conduct an evaluation by comparing the newly built corpus with our two other Japanese corpora and Sharoff's corpora.

Although a considerable amount of work has been done on ways to use the Web as a source of linguistic data, there are only few studies that have evaluated Web corpora, see e.g., for qualitative analyses, Fletcher (2004), Sharoff (this volume), Ueyama and Baroni (2005). Fletcher (2004) constructed a corpus of English via

automated queries to the AltaVista engine for the 10 top frequency words from the British National Corpus (henceforth BNC) and applied various post-processing steps to reduce the "noise" in the data (duplicates, boilerplate, etc.). He compared the frequency of various n-grams in the Web-derived corpus and in the BNC, finding the Web corpus to be 1) more oriented towards the US than the UK in terms of institutions, place names and spelling; 2) characterized by a more interactive style (frequent use of first and second person, present and future tense); 3) permeated by information technology terms; 4) more varied (despite the fact that the Web corpus is considerably smaller than the BNC, none of the most common 5,000 words in the BNC were absent from the Web corpus, but not vice versa). Properties 2) and 4) challenge the view that Web data are less fit to linguistic research than a carefully balanced corpus of texts obtained in other ways.

Sharoff (this volume) uses an adapted version of the BootCaT tools to build Web-derived corpora for English, Russian and German. The corpora are constructed via automated Google queries for random combinations of frequent words extracted from existing corpora. He classifies 200 documents randomly selected from each corpus in terms of various characteristics, including the topic domains of each document, analyzed using the BNC classification system (with some adaptations). He finds that, in a comparison with the BNC, the English Web corpus is richer in exemplars belonging to the technical and applied science domains. He also compares word frequencies his Web corpora with reference corpora in English and Russian, and newswire corpora in English, Russian and German. His results show that the Web corpora are closer to the reference corpora than to the newswire corpora, also confirming Fletcher's findings about the Web being characterized by a more interactive style and more lexical variety.

In an already mentioned previous study (Ueyama and Baroni 2005), we qualitatively evaluated two Japanese Web corpora built

in 2004 and 2005 with the use of the BootCaT tools. These are the corpora that here we call Genki 2004 and Genki 2005: see section 2 for details. The analysis showed that both corpora contained many documents produced by non-professional writers, characterized by everyday life topics and by an often informal, spontaneous, interactive style. Compared to Sharoff's results, we see that this text type is more dominant in our Japanese corpora than in any of his corpora in English, German, and Russian. We suspect that this difference between Sharoff's corpora and ours, i.e., a higher proportion of personal, spontaneous, interactive text in the latter, may be due to differences in seed choice. Our seeds, having been extracted from a basic vocabulary list from a Japanese textbook, are more often related to everyday life domains. In contrast, Sharoff's seeds are picked from existing traditional corpora (e.g., the BNC), and thus they tend to reflect some of the domains well represented in these corpora that are also common on the Web.[1]

The difference between Sharoff's and our results leads us to ask how different seed selection strategies affect the nature of resulting Web-based corpora. This is investigated by Ciaramita and Baroni (this volume) in a quantitative way. In this study, we perform a qualitative investigation, building and analyzing Japanese Web corpora using as seeds both words from a basic Japanese vocabulary list and words from Sharoff's English word list (based on the BNC) translated into Japanese. We conduct a relatively in-depth evaluation of the two resulting corpora in terms of domains, genres and typical lexical items, and discuss our findings in an attempt to answer the research question just described.

Another essential factor that affects Web corpus construction is time interval. It is well known that search engine indexing contin-

[1] A difference in the nature of the English and Japanese Webs, however, should not be completely ruled out, given a recent survey that indicates that the absolute number of blogs in Japanese is higher than the number of blogs in English. See `http://www.sifry.com/alerts/archives/000433.html`

uously changes, which is expected to strongly affect query results, and, consequently, the resulting Web corpus. The second goal of the study is therefore to investigate the effect of time interval and attempt to tackle the important issue of how "stable" the results of search engine queries are over time. For this purpose, we compare two Japanese Web corpora that we built at 10 months' distance from each other (in July 2004 and April 2005, respectively) with the use of exactly the same automated procedure and seeds. As for the investigation of the effects of seed selection, we analyze the distributions of domains, genres and typical lexical items in each corpus.

The rest of the paper is structured as follows. In section 2, we present the procedure used to build our three Japanese Web-based corpora (Genki 2004, Genki 2005, BNC-seeded 2005) and describe the characteristics of each corpus briefly. In section 3, we describe our corpus classification methods and present our results, while section 4 presents the evaluation of typical lexical items for each of the three corpora. Finally, in section 5 we discuss our findings and conclude by suggesting directions for further study.

2 Corpus construction

In this section, we describe our three Japanese Web corpora. We built the first two corpora with the same automated procedure and seed terms, but at two different times: the Genki 2004 corpus in July 2004, and the Genki 2005 in April 2005 (these were the corpora analyzed in Ueyama and Baroni 2005). The BNC-seeded 2005 corpus was built in August 2005, using the same procedure but different seeds.

For the Genki 2004 and 2005 corpora, in order to look for pages that were reasonably varied and not excessively technical, we considered that we should query a search engine (Google in our case) for words belonging to the basic Japanese vocabulary. Thus, we

randomly picked 100 words from the word list of *Genki*, an elementary Japanese Textbook (Banno et al. 1999; hence the name of the corpora): e.g., *tenki* "weather", *asagohan* "breakfast", *suupaa* "supermarket", *tsumetai* "cold". For the BNC-seeded 2005 corpus, we randomly picked 100 words from the list of 500 query terms that Sharoff extracted from the BNC to build his English Web corpus,[2] and translated those words into Japanese. The seeds that were selected for the construction of the BNC-seeded 2005 corpus vary more greatly in terms of domains (that include society, politics, history, computer technology) than the ones used for the two Genki corpora, that are very basic. We coherently translated the dictionary form of English verbs and adjectives into the dictionary form of their Japanese equivalents, although it is possible in theory to choose non-dictionary forms for Japanese translation candidates (e.g., formal present tense forms). In case both non-loanword and loanword varieties are available in Japanese, we employed the one that seems to be more common, which was expected to help to increase query hits: e.g., we translated "pattern" into *pataan* (loanword alternative), not *mohan* or *kata* (non-loanword alternatives).

All three Japanese Web corpora were built using the BootCaT tools mentioned earlier (Baroni and Bernardini 2004). We randomly combined the 100 seed terms into 100 triplets, and we used each triplet for an automated query to Google via the Google APIs (`http://www.google.com/apis`). The rationale for combining the words was that in this way we were more likely to find pages that contained connected text (since they contained at least 3 content-rich words). We used the very same triplets both in July 2004 and in April 2005 (for the Genki 2004 and 2005 corpora, respectively), while we created and used a new set of 100 triplets in August 2005 (for the BNC-seeded 2005 corpus). For each query, we retrieved maximally 10 URLs from Google, and we discarded

[2] `http://corpus.leeds.ac.uk/internet/seeds-en`

duplicate URLs. This gave us a total of 894 unique URLs in June 2004, 993 in April 2005, and 908 URLs in August 2005. Notice that, while for the purposes of our qualitative evaluation we are satisfied with corpora of these sizes, the same procedure could be used to build much larger corpora.

We compared the Genki 2004 and 2005 corpora in order to find how many URLs are present in both corpora. Interestingly, only 187 URLs were found in both, leaving 707 URLs that were retrieved in the Genki 2004 only and 806 URLs that were retrieved in the Genki 2005 only. Thus, with respect to the Genki 2005 URL list, the overlap with the previous year is of less than 20%. Moreover, there is of course no guarantee that the webpages corresponding to overlapping URLs between the two corpora did not change in terms of contents. To quickly investigate this point, we randomly selected 20 out of the 187 URLs retrieved in both years, and compared the 2004 and 2005 texts. We found that the two versions were identical in terms of contents for only 13 of the 20 URLs (65%), while the remaining pages had been modified (mostly for content updates). The changes in retrieved pages raise the question of whether the retrieved corpora are also different in terms of the nature of their contents or whether they are essentially comparable. This question will be examined later in section 3, on the basis of the results of the genre classification analysis. The overlap of URLs decreases even more between the Genki 2004 and BNC-seeded 2005 corpora. Only 11 URLs were present in both corpora. With respect to the Genki 2005 URL list, the overlap of URLs is only 1%.

For each URL, we (automatically) retrieved the corresponding webpage and formatted it as text by stripping off the HTML tags and other "boilerplate" (using Perl's `HTML::TreeBuilder` module and simple regular expressions). Since Japanese pages can be in different character sets (in particular, shift-jis, euc-jp, iso-2022-jp, utf-8), our script extracts the character set in which a page is

	total documents	total tokens	average size	error rate
Genki 2004	894	3,473,451	3,885	5%
Genki 2005	993	4,468,689	4,500	6%
BNC-seeded 2005	908	5,732,080	6,313	5%

Table 1. Total documents, total tokens, average size of tokens per document, and error rate in the Genki 2004, Genki 2005, and BNC-seeded 2005 corpora

encoded from the HTML code, and converts from that character set into utf-8. Since Japanese text does not use white space to separate words and characters, we used the ChaSen tool (Matsumoto at al 2000) to tokenize the downloaded corpora. However, ChaSen expects input and output to be coded in euc-jp, while our text-processing scripts are designed to receive text input coded in utf-8. To solve the problem of coding incompatibility, we used the `recode` tool[3] to convert back and forth between utf-8 and euc-jp.

According to the results of the ChaSen tokenization, the Genki 2004 corpus contains 3,473,451 tokens (about 3.5M); the Genki 2005 corpus 4,468,689 tokens (about 4.5M); the BNC-seeded 2005 corpus 5,732,080 tokens (about 5.7M).

Comparing the two Genki corpora, we have noticed that in Genki 2005 not only did the repeated queries find more and different URLs – they also found URLs that contained more text. This is illustrated by the average document size summarized in table 1. The BNC-seeded 2005 corpus, in turn, shows an increase of the total tokens of about 27%, and an increase of average document size of about 40% with respect to the Genki 2005 corpus, although the total document count decreases. We discuss the issue of the apparent trend of increase in corpus size and average document size in section 3, where the results of the corpus classification analysis are presented. We found (manually) that some pages did not contain any substantial amount of text: e.g., the ones that were not

[3] `http://recode.progiciels-bpi.ca/`

decoded properly, the ones that contained a warning message only, duplicates that were not removed, and so on. The ratio of these types of pages was approximately 5% for all the three corpora. We consider that this error rate is tolerable in the sense that the wide majority of text is usable.

3 Corpus classification

For the qualitative evaluation of our Japanese Web corpora, we manually classified all 894 pages of the Genki 2004 corpus, and 300 randomly selected pages each from the Genki 2005 and BNC-seeded 2005 corpora, in terms of topic domains and genre types.

3.1 Classification systems

3.1.1 Domains

For the classification of webpage domains, we adopted the classification system proposed in Sharoff (this volume), so that our results are directly comparable to his. We used the following nine categories:

natsci agriculture, astronomy, meteorology, ...

appsci computing, engineering, medicine, transport, ...

socsci law, history, sociology, language, education, religion...

politics

business e-commerce pages, company homepages, ...

life general topics related to everyday life typically for fiction, diaries, essays, etc...

arts literature, visual arts, performing arts, ...

leisure sports, travel, entertainment, fashion, hobbies ...

error encoding errors, duplicates, pages with a warning message only, empty pages

If a topic seemed to belong to more than one domain, we just selected one trying to be coherent. For example, we classified the webpages dedicated to a specific personal interest into the leisure domain, although the personal interests themselves are often related to everyday life, which is classified as the life domain (e.g., cooking, pets, etc.).

3.1.2 Genres

Webpages contain various genre types, including some attested in traditional corpora, e.g., news and diaries, and some newly emerging in Internet use, e.g., blogs (see Santini 2005). The situation is complicated by the fact that some documents can be a mix of more than one genre type (e.g., news report with an interactive discussion forum). Under these circumstances, it is not a simple task to classify Web documents by genre types. For the current study, the author first went through a good amount of the webpages to get a general idea of the distribution of genre types, and then selected the following 27 genre types as the final set:

blog personal pages created by users registered at blog servers that provide a ready-made page structure that, typically, include a diary with a comment section

BBS bulletin board sites; interactive discussion pages where multiple users can exchange messages with a topic-comments structure

diary a good example of an "adaptive" genre type that also exists in traditional written texts (see Santini 2005)

personal personal homepages not created through a blog service; less interactive than blogs since there is no interactive comment section

argessay essays written in an argumentative rhetoric style that present opinions, typically, on political or social issues

essay pages that state personal experiences, interests, feelings in a non-argumentative manner

novel another example of an adaptive genre type

commerinfo pages that present information to promote services or sell products

instrinfo pages designed to help readers to perform a certain task (how-to guides, guidelines, tips...)

info pages that present information that pertain to initiatives, events, resources and projects related to a certain topic without commercial or educational purposes (e.g., time/place of an upcoming event, political party manifestos, introduction to some academic program. . .)

teaching materials for instruction, typically, language teaching (e.g., example sentences, language exercises, ...)

news journalistic news; another adaptive genre type

njnews non-journalistic news, such as community pages

magazine Web magazine

acreport reports of academic research

report reports that present contents that pertain to a certain topic

review product/service evaluation, critique of arts, music, literature, etc.

comments comments directly sent from Web users, typically to commercial pages

questionnaire presentations of results of questionnaires

QA Q&A, FAQ, ...

list lists of words, numbers, etc

links lists of links to webpages with simple descriptions

top "top" pages that typically present the menu/structure of sites

speech speech or interview transcripts

errors pages that are not readable due to encoding problems, duplicates of other retrieved pages in the same corpus, pages with no contents

others cover class for genres represented by very few documents

Note that we broke down information and essay into subcategories depending on rhetorical types (i.e., argumentative, instructional etc.), being inspired in part by Santini (2005). We also distinguished journalistic from non-journalistic news, e.g., school or community news (news and njnews, respectively), and academic reports from non-academic ones (acreport and report, respectively). Finally, note the difference between info and report: the former pertains to information about a certain topic, e.g., information about some concert (the time and place of the event, etc.), while the latter presents contents that directly pertain to the topic, e.g., a report that presents the experience of going to the concert. We originally used more than the 27 classes reported above, but for ease of post-classification analysis, we collapsed categories with less than 3 pages in any corpus into the *others* category.

3.2 Results: Domains

3.2.1 Effects of time interval: Genki 2004 vs. Genki 2005

Since the Genki 2004 and 2005 corpora were constructed with the same procedure and with the same seed terms, but at different times (June 2004 and April 2005, respectively), the comparison of the two Genki corpora in terms of distribution of topic domains allows us to examine specifically how the time interval factor, which is 10 months in this case, affects the distribution of topic domains. The downloaded webpages were distributed across domains as shown in table 2, where the number and percentage of documents and their average size in number of tokens are summarized for each Genki corpus. The percentage values are also plotted in figure 1.

	Genki 2004			Genki 2005		
	# of docs	%	avg. size	# of docs	%	avg. size
appsci	24	2.7	2,451	8	2.7	3,914
arts	41	4.6	6,313	14	4.7	3,167
business	219	24.5	2,564	53	17.7	2,245
error	47	5.3	4,522	18	6	13,396
leisure	185	20.7	3,706	68	22.7	3,557
life	284	31.8	4,586	109	36.3	4,611
natsci	10	1.1	3,328	1	0.3	1,640
politics	7	0.8	5,826	1	0.3	1,573
socsci	77	8.6	4,151	28	9.3	8,564
total	894	100	3,885	300	100	4,744

Table 2. Distribution of topic domains in the Genki 2004 and 2005 corpora

Here we see that in both corpora life, business and leisure are the three major domain types, although there is a difference in ranking: life > business > leisure in 2004; life > leisure > business in 2005. This suggests an increase in the proportion of "personal interest" pages. The other domains are distributed in a more or less similar manner in the two corpora, as shown in figure 1. Some differences are found between the two Genki corpora, but we conclude that the effect of time interval is not very strong, since the two corpora share major characteristics, i.e., overall dominance of "personal interest" and commercial pages.

Comparing our results with the ones of Sharoff (for corpora in English, Russian, German), we notice that the total percentage of socsci and politics is only about 10% in our corpora, while his corpora overall show higher percentages, ranging from 15% to 29% in the three languages. Another difference is that our Genki corpora show a higher percentage of documents about life and leisure that refer to everyday life topics or personal interests. In our Genki corpora, the sum of life and leisure is consistently higher than 50% (52.5% in 2004, 59% in 2005), while in Sharoff's corpora the value ranges from 25% (English) to 51% (Russian). We suspect that these two differences between Sharoff's corpora and

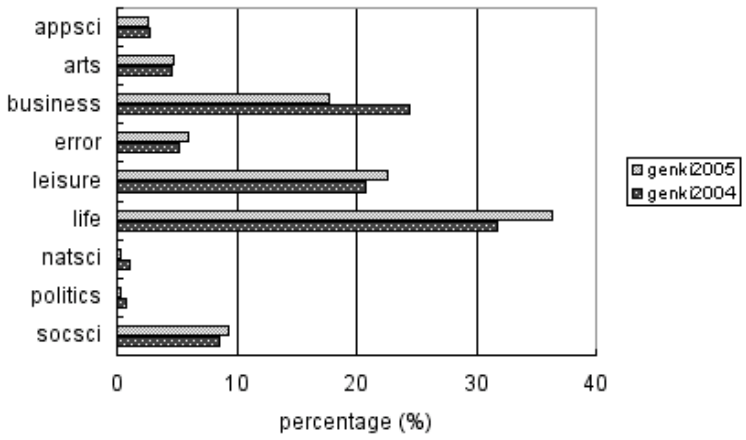

Figure 1. Percentage distribution of topic domains in the Genki 2004 and 2005 corpora

our corpora are mainly due to differences in seed choice. Our seeds, having been extracted from a basic vocabulary list, are more often related to everyday life domains, whereas Sharoff's seeds come from existing traditional corpora, and thus they tend to reflect some of the "higher" domains attested in these corpora. In the next section, we will investigate effects of seed selection by comparing the distribution of topic domain types in the Genki 2005 and BNC-seeded 2005 corpora.

3.2.2 Effects of seed selection: Genki 2005 vs. BNC-seeded 2005

The distribution of topic domain types is summarized in table 3, where the number and percentage of documents and their average size in number of tokens are presented for each domain type for the Genki 2005 and BNC-seeded 2005 corpora. The percentage values are also plotted in figure 2. Genki 2005 and BNC-seeded 2005 show more differences in domain distributions than the two

	Genki 2005			BNC-seeded 2005		
	# of docs	%	avg. size	# of docs	%	avg. size
appsci	8	2.7	3,914	17	5.7	3,702
arts	14	4.7	3,167	15	5	8,469
business	53	17.7	2,245	75	25	2,465
error	18	6	13,396	15	5	4,480
leisure	68	22.7	3,557	36	8.7	7,684
life	109	36.3	4,611	30	10	6,813
natsci	1	0.3	1,640	21	7	2,957
politics	1	0.3	1,573	65	21.7	6,037
socsci	28	9.3	8,564	36	12	7,103
total	300	100	4,744	300	100	5,188

Table 3. Distribution of topic domains in the Genki 2005 and BNC-seeded 2005 corpora

Genki corpora. With respect to Genki 2005, the proportions of five topic domains, appsci, business, natsci, politics, socsci – and the latter two in particular – are much higher than in BNC-seeded 2005. A decrease in leisure and life appears to be a trade-off of this increase. These differences cue two general changes that are likely to be caused by the change of seeds: an increase in the proportion of scientific and socio-political pages, and a decrease in the proportion of "personal interest" pages.

Strictly speaking, the comparison of Genki 2005 and BNC-seeded 2005 is not the best way of investigating effects of seed selection by excluding effects of time interval, since the two corpora were not constructed at the same time: the Genki 2005 corpus was built in April 2005, the BNC-seeded 2005 in August 2005. However, considering that the differences between the two corpora (at a 4-month interval) are much greater than those between the two Genki corpora (at a 10-month interval), we believe it is safe to conclude that the distribution of topic domain types in a Web corpus depends more on seed selection than on time interval.

Comparing our BNC-seeded 2005 corpus with Sharoff's En-

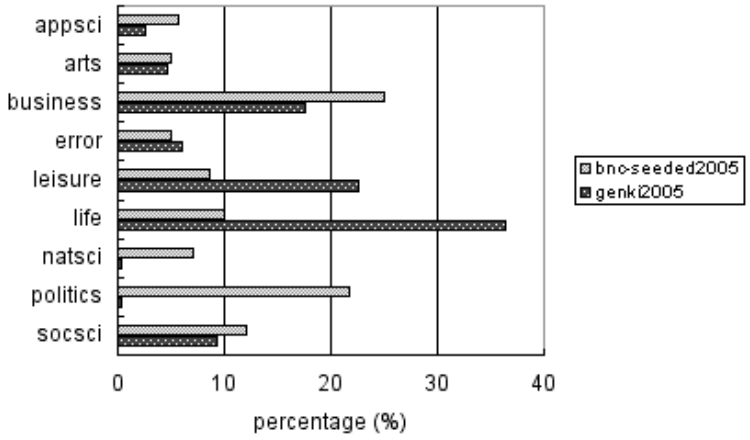

Figure 2. Percentage distribution of topic domains in the Genki 2005 and BNC-seeded 2005 corpora

glish Web corpus it is appropriate to examine similarities and differences between English and Japanese in the distribution of domain types. The reasoning is that the two corpora were built with more or less the same automated procedure and with similar seeds (we picked 100 words randomly from Sharoff's English word list), although, again, the corpora were not constructed at the same time, and, of course, there may also be effects due to the difference in annotators. The percentage distribution of domain types in our Japanese corpus (BNC-seeded 2005) and his English corpus (I-EN) is presented in figure 3. There are several notable differences. Two major domains in BNC-seeded 2005 are business and politics, as opposed to appsci and socsci in the I-EN corpus. Sharoff reported that in the I-EN corpus the majority of socsci pages are legal texts (legislation, law reports, terms and conditions, etc.), but we found almost no case of legal text in the BNC-seeded 2005, where a majority of pages labeled as socsci belong to other subdomains such as sociology, education or

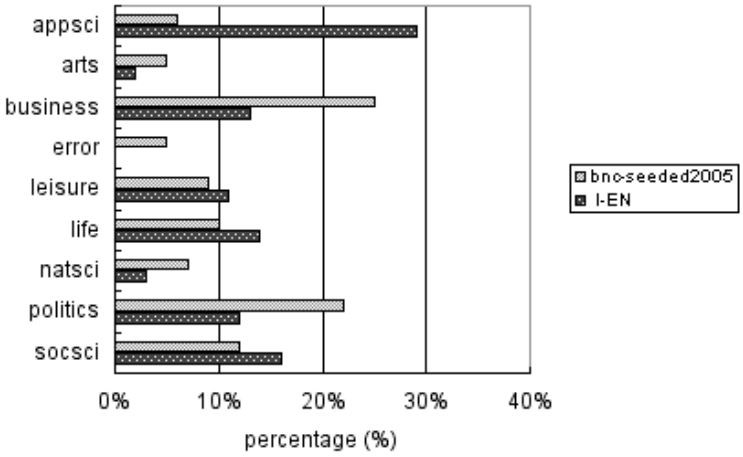

Figure 3. Percentage distribution of domain types in Sharoff's English corpus (I-EN) and our Japanese corpus (BNC-seeded 2005)

language. For the other domain types, we found no obvious difference. These results suggest that Web documents in different languages (at least, English and Japanese as indexed by Google) differ in the distribution of topic domains.

3.3 Results: Genre types

3.3.1 Effects of time interval: Genki 2004 vs. Genki 2005

The distribution of genre types in the two Genki corpora is presented in table 4, which summarizes the number and percentage of documents and their average size in number of tokens for each genre type. The percentage values are also plotted in figure 4. The general pattern that we found here is that in both corpora the genre types typical of personal prose – i.e., BBS, blog, diary, essay and personal – occupy a good portion of the distribution. The sum of these genres is 39.9% in Genki 2004 and 49% in Genki 2005. The overall dominance of the personal genres indicates that

the Web-based corpora are likely to include a good amount of spontaneous prose produced by non-professional writers, which seems to match the dominance of "personal interest" pages in the results of the domain evaluation of the Genki corpora presented in section 3.2.1. Since this type of prose is not available in traditional corpora, Web-based corpora can be a very precious new linguistic resource.

Interestingly, we notice a sharp increase in the overall proportion of these genres between 2004 and 2005, suggesting the possibility that the Japanese Web (at least as ranked by Google and retrieved with our method) is becoming richer in personal prose. Another prominent genre type is commerinfo (commercial information). It occupies 18.6% and 14% of Web documents in the Genki 2004 and 2005 corpora, respectively (indicating that, at least according to our sample, its overall share is receding, perhaps in correspondence with the increase in personal pages). Together, personal and commercial pages constitute the majority of our Web-based corpora. The sum of these two types is 58.5% and 63% in 2004 and 2005, respectively. In contrast, the ratio of news is surprisingly low (1.1% in 2004, 0% in 2005), and there is no single case of acreport (reports of academic research) in either corpus. This may again be caused by our selection of seed terms, as was probably the case for the low percentage of politics and socsci in the results of the domain evaluation of the Genki corpora.

The genre types that tend not to include a good chunk of prose, such as links (links to other webpages), top (top pages with a site menu) and list (lists of words or numbers), have a relatively low ratio (8.6% in 2004 and 5.6% in 2005 in total). This is, of course, good news.

In summary, the genre evaluation of the Genki 2004 and 2005 corpora shows that a good majority of Web documents retrieved with Genki seeds are constituted by personal or commercial genres rather than academic or journalistic genres, which fits in nicely

	Genki 2004			Genki 2005		
	# of docs	%	avg. size	# of docs	%	avg. size
acreport	0	0	0	0	0	0
argessay	7	0.8	3,158	4	1.3	3,524
BBS	54	6.0	8,243	10	3.3	9,329
blog	55	6.2	3,959	74	24.7	4,604
comments	10	1.1	2,040	9	3.0	7,248
commerinfo	166	18.6	2,433	42	14.0	2,393
diary	165	18.5	5,019	47	15.7	5,284
error	51	5.7	4,171	18	6.0	13,396
essay	66	7.4	3,414	12	4.0	4,897
info	14	1.6	1,813	8	2.7	2,296
instinfo	32	3.6	2,790	9	3.0	3,588
links	48	5.4	1,768	7	2.3	2,327
list	15	1.7	4,949	6	2.0	550
magazine	13	1.5	4,332	0	0	0
news	10	1.1	3,316	0	0	0
njnews	5	0.6	5,109	3	1.0	1,426
novel	18	2.0	10,367	4	1.3	3,236
others	10	1.1	4,207	8	2.7	7,780
personal	16	1.8	2,138	4	1.3	1,909
QA	33	3.7	2,966	4	1.3	2,759
questionnaire	24	2.7	3,724	5	1.7	1,393
report	51	5.7	2,367	15	5.0	3,492
review	5	0.6	5,733	0	0	0
speech	5	0.6	9,131	4	1.3	2,671
teaching	8	1.9	5,362	3	1.0	3,741
top	13	1.5	1,623	4	1.3	2,893
total	894	100	3,885	300	100	4,744

Table 4. Distribution of genre types in the Genki 2004 and 2005 corpora

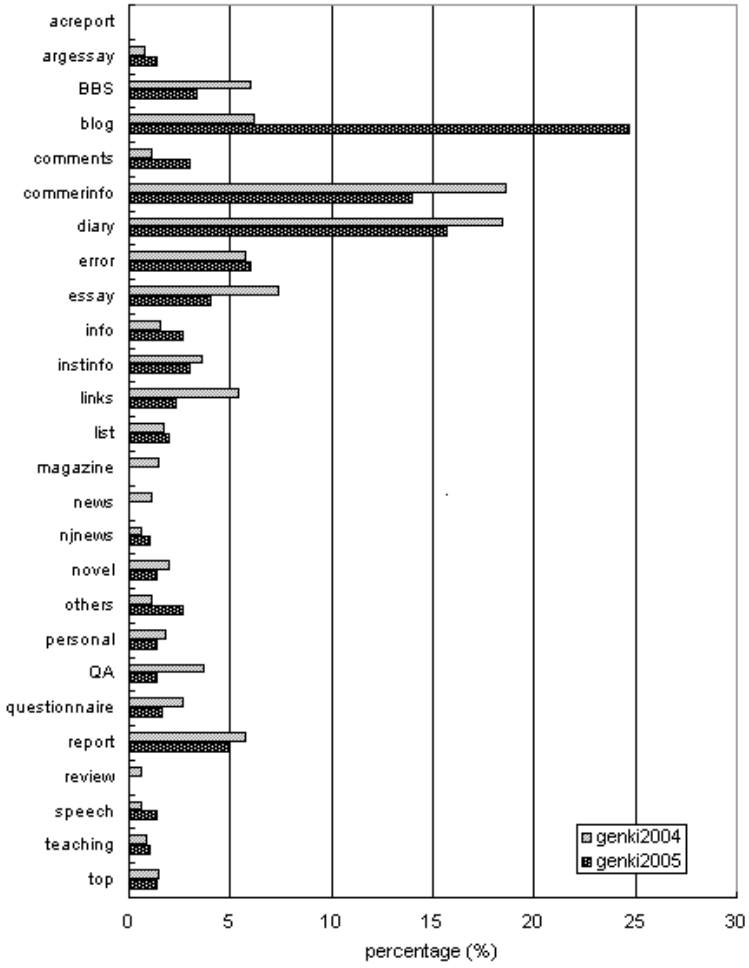

Figure 4. Percentage distribution of genre types in the Genki 2004 and 2005 corpora

with the results of the domain classification. This overall pattern is observed commonly in both corpora, although there are some differences, e.g., an increase in the proportion of personal genres, which suggests that the Japanese Web may be becoming richer in personal prose.

3.3.2 Effects of seed selection: Genki 2005 vs. BNC-seeded 2005

We also compared the Genki 2005 and BNC-seeded 2005 corpora in terms of the distribution of genre types in order to further examine effects of seed selection. The results of the genre evaluation are presented in table 5 and figure 5. Here we find some dramatic changes between the two corpora. In the BNC-seeded 2005, there is a sharp decrease in the proportion of pages of blog and diary, two major personal genres, while there is a substantial increase in the proportion of genres where academic, journalistic or public contents are presented (e.g., acreport, argessay, news and report). These changes in genre distribution match with the results of the domain evaluation that show an increase in the proportion of scientific and sociopolitical topics.

We notice that the magnitude of the changes between the Genki 2005 and BNC-seeded 2005 corpora in the genre type distribution is much greater than that between the two Genki corpora. Considering this finding, we believe that the distribution of genre types in the Web corpus largely depends on the nature of seed selection just like in the case of the distribution of domain types (see section 3.2).

3.4 Discussion

We have manually classified webpages of our three Japanese Web corpora in terms of domains and genres to examine how time interval and seed selection affect characteristics of the resulting Web

	Genki 2005			BNC-seeded 2005		
	# # of docs	%	avg. size	# of docs	%	avg. size
acreport	0	0	0	8	2.7	11,172
argessay	4	1.3	3,524	25	8.3	4,916
BBS	10	3.3	9,329	4	1.3	19,757
blog	74	24.7	4,604	19	6.3	7,228
comments	9	3.0	7,248	3	1.0	1,325
commerinfo	42	14.0	2,393	48	16.0	1,693
diary	47	15.7	5,284	16	5.3	8,079
error	18	6.0	13,396	15	5.0	4,480
essay	12	4.0	4,897	11	3.7	6,179
info	8	2.7	2,296	21	7.0	3,325
instinfo	9	3.0	3,588	10	3.3	3,324
links	7	2.3	2,327	0	0	0
list	6	2.0	550	5	5	7,876
magazine	0	0	0	12	4	8,039
news	0	0	0	13	4.3	6,065
njnews	3	1.0	1,426	4	1.3	5,418
novel	4	1.3	3,236	5	1.7	14,522
others	8	2.7	7,780	9	3.0	3,868
personal	4	1.3	1,909	18	6.0	6,517
QA	4	1.3	2,759	0	0	0
questionnaire	5	1.7	1,393	0	0	0
report	15	5.0	3,492	36	12.0	3,320
review	0	0	0	0	0	0
speech	4	1.3	2,671	6	2.7	4,248
teaching	3	1.0	3,741	4	2.0	348
top	4	1.3	2,893	8	1.3	11,172
total	300	100	4,744	300	100	5,188

Table 5. Distribution of genre types in the Genki 2005 and BNC-seeded 2005 corpora

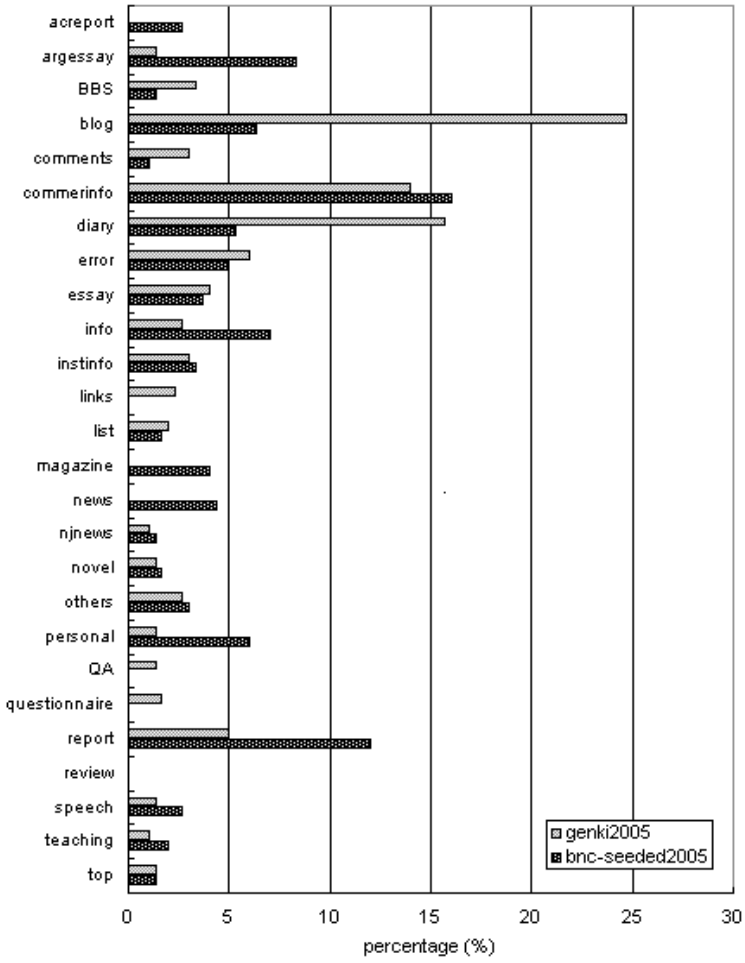

Figure 5. Percentage distribution of genre types in the Genki 2005 and BNC-seeded 2005 corpora

corpora. The two main findings have been as follows: 1) both factors affect characteristics of Web corpora considerably; 2) however, the effect of seed selection is notably stronger than that of time. In consideration of the results of corpus classification, one might wonder if the general increase in corpus size and average document size both from Genki 2004 to Genki 2005 and from Genki 2005 to BNC-seeded 2005, which was reported in section 2, are due to differences in the domains/genres that characterize the various corpora. We thoroughly examined the distributions of sizes within domains and genres for each pair (Genki 2004 vs. 2005, and Genki vs. BNC-seeded 2005), but we did not find any systematic correlation between the average text size and the distribution patterns of domains and genres. This indicates that the general increase of the average corpus size is not caused by changes in distribution of text types in a systematic way. One possible alternative explanation is that a good number of webpages increases in size over time as new contents are added. It will be interesting to examine this possibility by observing chronological changes in text size for the same webpages.

4 Typical lexical items

In this section, we examine how time interval and seed selection affect Japanese Web corpus construction from a lexical point of view. For this purpose, we conducted a qualitative analysis of typical lexical items in our three Japanese Web corpora. For two pairs of our three Japanese Web corpora (Genki 2004 vs. 2005 and Genki vs. BNC-seeded 2005), we compared the frequency of occurrence of each "word" (as tokenized by ChaSen) in each corpus with its frequency in the other corpus by computing the log-likelihood ratio association measure (Dunning 1993). We then evaluated the lists of words ranked by log-likelihood ratio, focusing in particular on the top 300 items in each list (Sharoff applies the same

methodology; see his article for a discussion of the log-likelihood ratio measure).

In the top lists of the two Genki corpora, we did not find any systematic difference except for the following. The Genki 2004 list contains more lexical items related to business or finance (e.g., *tenpo* "store", *gokakunin* "confirmation") – 29 relevant items in the top 300 list – while there are only 3 items in the top 300 list of the Genki 2005. This may be explained by the higher proportion of pages classified as business in Genki 2004 than in Genki 2005, as reported earlier. In contrast, some dramatic difference has emerged from the comparison of the top 300 word lists of the Genki 2005 and BNC-seeded 2005 corpora. The BNC-seeded 2005 list contains a high proportion of terms used in socio-political text, i.e., 43% of the list (e.g., *seefu* "government", *kenpoo* "constitution"), while no instance of this sort is found in the Genki 2005 list. The difference must be due to the change in seed selection that has caused a major boost in the proportion of socio-political text.

In summary, the analysis of the data ranked by log-likelihood ratio for the Genki 2004 and 2005 corpora did not show any fundamental differences, while a strong difference emerged from the results of the comparison between the Genki 2005 and BNC-seeded 2005 corpora. This indicates that seed selection impacts on the lexical distribution of the resulting corpus more than time interval, as it does with the composition of domains and genres (the phenomena are obviously related).

5 Conclusion

The qualitative evaluation of the Japanese Web corpora built with automatic queries to Google coherently shows the following two patterns: 1) both seed selection and time interval affect the distribution of text and lexicons in the resulting Web corpus; 2) the effect of seed selection is much stronger than the effect of time

interval. The difference between the two examined factors in magnitude of effects may be partly explained by the fact that the two factors affect Web-based corpus construction in different ways. Seed selection directly pertains to the way in which we sample documents from the Web. However, this is not the case for time interval. Time interval is rather relevant to changes in extrinsic factors such as indexing and ranking of Web documents by search engines, modifications of webpage contents, and so on. Such extrinsic factors largely characterize the dynamic nature of Web documents, but the changes due to time interval between corpus construction sessions affect the overall distributional properties of the resulting Web-based corpora, in terms of domain, genre and lexicon, much less than seed selection. To further study this point, we would like to observe chronological changes by repeatedly constructing Web-based corpora with a certain fixed time interval and the same procedure used to build Genki 2004 and 2005.

The prominent effect of seed selection on Web corpus construction suggests that a good understanding of the cause-and-effect relation between seeds and retrieved documents is an important step to gain some control over the characteristics of Web-based corpora, in particular, for the construction of general-purpose or reference corpora that are meant to represent a language as a whole. This boils down to a need to understand distributional properties of Web documents and then find a good method to randomly sample a set of documents that represent those properties with minimal bias toward certain domains, and seed selection is a very crucial part of the automatic sampling process. As far as we know, this line of research has not been widely pursued yet, except for the preliminary experiments by Ciaramita and Baroni (this volume). They propose and test an automated, quantitative, knowledge-poor method to evaluate the randomness of a Web corpus (with respect to a number of non-random/biased partitioning of the whole collection of Web documents). The results of their

experiments indicate some effect of seed frequency on the randomness of the resulting corpus: i.e., medium frequency seeds might lead to a less biased corpus than either high frequency terms or terms selected from the whole frequency range. This line of research is crucial for finding an effective automated method to construct general-purpose balanced corpora from the Web. We are interested in further testing the effect of different seed sets picked on the basis of frequencies, and semantic/topical domains (e.g, arts, leisure, life, politics, etc.), to see how the properties of seed sets correlate with the distributional properties and quality of the resulting corpus.

References

Banno, E., Onno, Y., Sakane, Y. and Shinagawa, C. (1999). *Genki: An integrated course in elementary Japanese*. Tokyo: The Japan Times.

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. In *Proceedings of LREC 2004*, 1313-1316.

Baroni, M. and Ueyama, M. (2004). Retrieving Japanese specialized terms and corpora from the World Wide Web. *Proceedings of KONVENS 2004*.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics* 19(1), 61-74.

Fetterly, D., Manasse, M., Najork, M. and Wiener, J. (2004) A large-scale study of the evolution of Web pages. *Software: Practice & Experience* 34, 213-237.

Fletcher, B. (2004). Making the Web more useful as a source for

linguistic corpora. In Connor, U. and Upton, T. (eds.) *Corpus linguistics in North America 2002*, Amsterdam: Rodopi.

Ghani, R., Jones, R.and Mladenić, D. (2003). Building minority language corpora by learning to generate Web search queries. *Knowledge and Information Systems* 7(1), 56-83.

Goto, H. (2003). Linguistic theories and linguistic resources: corpora and other data (Gengo riron to gengo shiryoo: coopasu to coopasu igai no deeta). *Nihongogaku (Japanese Language Studies)* 22, 6-15.

Kilgarriff A. and Grefenstette, G. (2003). Introduction to the special issue on the Web as Corpus. *Computational Linguistics* 29(3), 333-347.

Matsumoto, Y., Kitauchi, A., Yamashita, T., Hirano, Y., Matsuda, H., Takaoka, K., and Asahara, M. (2000). Morphological analysis system ChaSen version 2.2.1 manual. NIST Technical Report.

Santini, M. (2005). Genres in formation? An exploratory study of Web pages using cluster analysis. Proceedings of CLUK 05.

Ueyama, M. and Baroni, M. (2005). Automated construction and evaluation of a Japanese Web-based reference corpus. *Proceedings of Corpus Linguistics 2005*, available online at `http://www.corpus.bham.ac.uk/PCLC/`.

Measuring Web Corpus Randomness: A Progress Report

Massimiliano Ciaramita and Marco Baroni

1 Introduction

The Web is a very rich source of linguistic data, and in the last few years it has been used very intensively by linguists and language technologists for many tasks (see Kilgarriff and Grefenstette 2003 for a review of some of the relevant work). Among other uses, the Web allows fast and inexpensive construction of "reference"/"general-purpose" corpora, i.e., corpora that are not meant to represent a specific sub-language, but a language as a whole. There is a large literature on the issue of representativeness of corpora (see, e.g., Biber 1993), and several recent studies on the extent to which Web-derived corpora are comparable, in terms of variety of topics and styles, to traditional "balanced" corpora (e.g., Fletcher 2004, Sharoff this volume). Our contribution, in this paper, is to present an automated, quantitative method to evaluate the "variety" or "randomness" (with respect to a number of non-random partitions) of a Web corpus. The more random/less biased towards a specific partition a corpus is, the more it should be suitable as a general-purpose corpus. It is important to realize that we are not proposing a method to evaluate whether a sample of webpages is a random sample of the Web. Instead, we are proposing a method to evaluate if a sample of webpages in a

certain language is reasonably varied in terms of the topics (and, perhaps, textual types) it represents.

In our evaluation of the method, we focus on general-purpose corpora built issuing automated queries to a search engine and retrieving the corresponding pages, which has been shown to be an easy and effective way to build Web-based corpora (see section 2 below). With respect to this approach, it is natural to ask which kinds of query terms (henceforth seeds) are more appropriate to build a corpus that is comparable, in terms of variety and representativeness, to a traditional balanced corpus such as the British National Corpus (BNC). We will test our method for assessing Web corpus randomness on corpora built with low, medium and high frequency seeds. However, the method *per se* can also be used to assess the randomness of corpora built in other ways (e.g., by crawling the Web starting from a few selected URLs).

Our method is based on the comparison of the word frequency distributions of the target corpus to word frequency distributions constructed using queries to a search engine for deliberately biased seeds (i.e., instead of trying to compare the corpus to a supposedly unbiased corpus, we look at how it compares to corpora that we are almost certain are highly biased). As such, it is nearly resource-free, as it only requires lists of words belonging to specific domains that can be used as biased seeds. While in our experiments we used Google as the search engine of choice, and in what follows we often use "Google" and "search engine" interchangeably, our procedure could also be carried out using a different search engine (or other ways to obtain collections of biased documents, e.g., via a directory of pre-categorized webpages).

After reviewing some of the relevant literature in section 2, in section 3 we introduce and justify our methodology. We show how, when we can sample randomly from the whole BNC and from its domain and genre partitions, our method to measure distance between sets of documents produces intuitively plausible results

(similar partitions are nearer each other), and that the most varied, least biased distribution (the one from the whole BNC) is the one that has the least average distance from all the other (biased) distributions (we provide a geometric explanation of why this is the case). Hence, we propose average distance from a set of biased distributions as a way to measure corpus randomness: the lower the average distance, the more random/unbiased the corpus is. In section 4, we apply our technique to unbiased and biased corpora constructed via Google queries. The results of the Google experiments are very encouraging, in that the corpora built with various unbiased seed sets show, systematically, significantly shorter average distance to the biased corpora than any corpus built with biased seeds. Among unbiased seed sets chosen from high and medium frequency words, and from the whole frequency range, medium frequency words appear to be the best (in the sense that they lead to the least biased corpus, according to our method). In section 5, we conclude by summarizing our main results, considering some open questions and sketching directions for further work.

2 Relevant work

Our work is obviously related to the recent literature on building linguistic corpora from the Web using automated queries to search engines (see, e.g., Ghani et al. 2001, Fletcher 2004, Baroni and Bernardini 2004, Sharoff this volume, Ueyama this volume). With the exception of Baroni and Bernardini, who are interested in the construction of specialized language corpora, these researchers use the technique to build corpora that are meant to function as general-purpose "reference" corpora for the relevant language.

Different criteria are used to select seed words. Ghani and colleagues iteratively bootstrap queries to AltaVista from retrieved documents in the target language and in other languages. They

seed the bootstrap procedure with manually selected documents, or with small sets of words provided by native speakers of the target language. They evaluate performance in terms of how many of the retrieved pages are in the relevant language, but do not assess their quality or variety. Fletcher constructs a corpus of English by querying AltaVista for the 10 top frequency words from the BNC. He then conducts a qualitative analysis of frequent n-grams in the Web corpus and in the BNC, highlighting the differences between the two corpora. Sharoff (this volume; see also Sharoff submitted) builds corpora of English, Russian and German using queries to the Google search engine, seeded with manually cleaned lists of words that are frequent in a reference corpus in the relevant language, excluding function words. Sharoff evaluates the results both in terms of manual classification of the retrieved pages and by means of a qualitative analysis of the words that are most typical of Web corpora vs. other corpora. For English, he also provides a comparison of corpora retrieved using non-overlapping but similarly selected seed sets, concluding that the difference in seeds is not having a strong effect on the nature of the pages retrieved. Ueyama (this volume; see also Ueyama and Baroni 2005) builds corpora of Japanese using both words from a basic Japanese vocabulary list, and translations from one of Sharoff's English lists (based on the BNC) as seeds. Through qualitative methods similar to those of Sharoff, she shows how the corpus built using basic vocabulary seeds is characterized by more "personal" genres than the one constructed from BNC-style seeds.

Like Sharoff and Ueyama, we are interested in evaluating the effect that different seed selection (or, more in general, corpus building) strategies have on the nature of the resulting Web corpus. However, rather than performing a qualitative investigation, we develop a quantitative measure that could be used to evaluate and compare a large number of different corpus building methods, as it does not require manual intervention. Moreover, our empha-

sis is not on the corpus building methodology, nor on classifying the retrieved pages, but on assessing whether they appear to be reasonably "unbiased" with respect to a range of topics or other criteria.

A different line of research somewhat related to ours pertains to the development of methods to perform quasi-random sampling of documents from the Web. The emphasis is not on corpus building, but on estimating statistics such as the percentage of pages in a certain domain, or the size and overlap of pages indexed by different search engines. For example, both Henzinger et al. (2000) and Bar-Yossef et al. (2000) use random walks through the Web, represented as a graph, to answer questions of this kind. Bharat and Broder (1998) issue random queries (based on words extracted from documents in the Yahoo! hierarchy) to various search engines in order to estimate their relative size and overlap. There are two important differences between work in this tradition and ours. First, we are not interested in an unbiased sample of webpages, but in a sample of pages that, taken together, can give a reasonably unbiased picture of a language, independently of whether they are actually representing what is out there on the Web or not. For example, although computer-related technical language is probably much more common on the Web than, say, the language of literary criticism, we would prefer a biased retrieval method that fetches documents representing these and other sub-languages in comparable amounts, to an unbiased method that leads to a corpus composed mostly of computer jargon. Second, while here we analyze corpora built via random queries to a search engine, the focus of the paper is not on this specific approach to Web corpus construction, but on the procedure we develop in order to evaluate how varied the linguistic sample we retrieve is. Indeed, in future research it would be interesting to apply our method to corpora constructed using random walks of the Web, along the lines of Henzinger, Bar-Yossef and their colleagues.

3 Measuring distributional properties of biased and unbiased collections

Our goal is to create a "balanced" corpus of webpages from the portion of the Web which contains documents of a given language; e.g., the portion composed of all Italian webpages. As we observed in the previous section, obtaining a sample of unbiased documents is not the same as obtaining an unbiased sample of documents. Thus, we will not motivate our method in terms of whether it favors unbiased samples from the Web, but in terms of whether the documents that are sampled appear to be balanced with respect to a set of deliberately biased samples. We leave it to further research to study how the choice of the biased sampling method affects the performance of our procedure. In this section, we introduce our approach by discussing experiments conducted on the BNC where the corpus is seen as a model for the Web, that is, a large collection of documents of different nature. We investigate the distributional properties of the BNC, and the known categories defined within the corpus, which are fully accessible and therefore suitable for random sampling. The method we present highlights important properties that characterize the overall distribution of documents inferrable from incomplete and noisy sampled portions of it; e.g., those which can be retrieved using a suitable set of seed words. In later sections we will show how the method works when the full corpus, the Web, is not available and there is no alternative to "noisy" sampling.

3.1 Collections of documents as unigram distributions

A compact way of representing a collection of documents is by means of a frequency list, where each word is associated with the number of times it occurred in the collection. This representation

defines a simple "language model", a stochastic approximation to the language used in the collection; i.e., a "0th order" word model or a "unigram" model. Language models of varying complexity can be defined. As the model's complexity increases, its approximation to the target language improves (cf. Shannon's classic example on the entropy of English – Shannon 1948). In this paper we focus on the unigram model as a natural starting point; however the methods we investigate extend naturally to more complex language models.

3.2 Similarity measures for document collections

Our method works by measuring the similarity of collections of documents, approximated as the similarity of the derived unigram distributions, based on the assumption that two similar document collections will determine similar language models. We experimented with two similarity measures over unigram models. The first is the *relative entropy*, or *Kullback Leibler distance* (also referred to as KL), $D(p||q)$ (cf. Cover and Thomas 1991, p. 18), defined over two probability mass functions $p(x)$ and $q(x)$:

$$D(p||q) = \sum_{x \in W} p(x) \log \frac{p(x)}{q(x)} \tag{1}$$

The relative entropy is a measure of the cost, in terms of average number of additional bits needed to describe the random variable, of assuming that the distribution is q when instead the true distribution is p. Since $D(p||q) \geq 0$, with equality only if $p = q$, unigram distributions generated by similar collections should have low relative entropy. KL is finite only if the support set of q is contained in the support set of p, hence we make the assumption that the random variables always range over the dictionary W, the set of all word types occurring in the BNC. To avoid infinite cases

Word	Unigram		Total
	P	Q	
w_1	33	17	50
w_2	237	156	393
..	..	..	..
$w_{\|W\|}$	26	1	27
Total	138,574	86,783	225,357

Table 1. Sample contingency table for two unigram distributions P and Q

a smoothing value α is added when estimating probabilities; i.e.,

$$p(x) = \frac{count_P(x) + \alpha}{|W|\alpha + \sum_{x \in W} count_P(x)} \quad (2)$$

where $count_P(x)$ is the frequency of x in the unigram distribution P, and $|W|$ is the number of word types in W.

Another way of assessing the similarity of unigram distributions is by analogy with categorical data analysis in statistics, where the goal is to assess the degree of *dependency*, or contingency, between two classification criteria. Given two distributions P and Q we create a contingency table in which each row represents a word in W, and each column represents, respectively, frequencies in P and Q (see table 1). If the two distributions are independent from each other, a cell probability will equal the product of its respective row and column probabilities; e.g., the probability that w_1 will occur in distribution P is $p(w_1) \times p(\mathrm{P}) = \frac{50}{225,357} \times \frac{138,574}{225,357} = 0.000135$. The expected number of times w_1 occur in P, under the null hypothesis that P and Q are independent, is then $e_{1,P} = N \times p(w_1)p(\mathrm{P}) = (225,357) \times (0.000135) = 30.48$, as in a multinomial experiment. If the hypothesis of independence is true then the observed cell counts should not deviate greatly from the expected counts. Here we use the X^2 (chi-square) test statistic, involving the $|W|$ deviations, to measure the degree of dependence

between P and Q, and thus – intuitively, their similarity:

$$X^2 = \sum_{i,j} \frac{[o_{i,j} - e_{i,j}]^2}{e_{i,j}} \quad (3)$$

Rayson and Garside (2000) use a similar approach to corpus comparison, where deviations in the use of individual words are compared. Here we compare distributions over the whole dictionary to measure the similarity of two text collections.

3.3 Similarity of BNC partitions

In this section we introduce and test the general method in a setting where we can randomly sample from the whole BNC corpus (a classic example of a "balanced" corpus, Aston and Burnard 1998) and from its labeled subsets. The BNC contains 4,054 documents composed of 772,137 different types of words with an overall frequency, according to our tokenization, of 112,181,021 word tokens. Documents come classified along different dimensions. In particular, we adopt here David Lee's revised classification (Lee 2001) and we partition the documents in terms of "mode" (spoken/written), "domain" (19 labels; e.g., imaginative, leisure, etc.) and "genre" (71 labels; e.g., interview, advertisement, email, etc.) For the purposes of the statistics reported below, we filter out words belonging to a stop list containing 1,430 types and composed mostly of function words. These were extracted in two ways: they either were already labeled with one of the function word tags in the BNC (such as "article" or "coordinating conjunction") or they occurred more than 50,000 times.

Relative entropy and chi-square intuitively measure how similar two distributions are. A simple experiment illustrates the kind of outcomes they produce. If the similarity between pairs of unigrams, corresponding to specific BNC genres or domains is measured, often the results match our intuitions. For example, in

S_meeting				
S_meeting	S_meeting			
R	Genre	KL	Genre	X^2
1	S_meeting	0	S_meeting	0
2	S_brdcast_discuss	0.27	S_interview	82,249
3	S_speech_unscript	0.39	S_parliament	97,776
4	S_unclassified	0.41	S_brdcast_doc	100,566
5	S_interview_hist	0.44	S_speech_unscript	103,843
..	..	..	..	
67	S_demonstration	1.45	W_ac_soc_science	914,666
68	W_fict_drama	1.48	W_pop_lore	973,534
69	S_lect_nat_sci	1.54	W_non_ac_pol_law	976,794
70	S_lect_commerce	1.61	W_misc	1,036,780
71	W_fict_pros	1.64	W_fict_prose	1,640,670

Table 2. Similarities and differences among genres

the case of the genre "S_meeting"[1] the 5 closest (and least close) genres are those listed in table 2.

The table shows that both measures rank higher genres which refer to speech transcriptions of situations involving several people speaking (discussions, interviews, parliament reports, etc.), as is the case with the transcriptions relative to the target category "S_meeting". On the other hand, at the bottom of the ranking, we find written literary texts, or transcriptions of prepared speeches, which are more dissimilar to the target genre.

Figure 1 plots the matrices of distances between unigrams corresponding to different BNC domains for both X^2 and KL; domains are ordered alphabetically on both x and y axis. Overall the two plots have a somewhat similar topology, resembling a double plateau with peaks on the background. The plot shows, not too surprisingly, that speech transcriptions (whose domain names are

[1] "S_" is the prefix for spoken categories, while "W_" is the prefix for written categories.

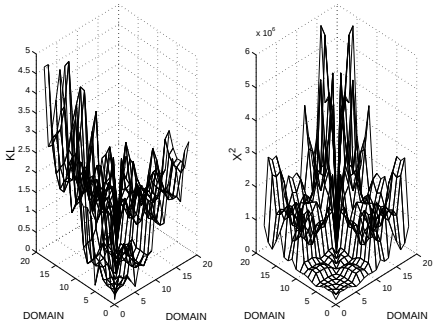

Figure 1. Plots of KL and X^2 distance matrices for the "domain" BNC partitions

prefixed with an "S") tend to be more similar to each other than to written text ("W"-prefixed domains), and vice-versa. However, the figure also shows several important differences between the measures. First of all, X^2 is symmetric while KL is not. In particular, if the size of the two distributions varies greatly, as between the first few domains (close to 1) and the last ones (close to 19) the choice of the background distribution in KL has an effect on the magnitude of the distance: greater if the "true" distribution is larger because of the log-likelihood ratio.

More important is the difference emerging from the region far in the background. Here the two measures give very different rankings. In particular, X^2 tends to interleave the rankings of written and spoken categories. X^2 also ranks lowest several written domains. Table 3 illustrates this fact with an example, where the target domain is "W_world_affairs". Interestingly, X^2 ranks low domains such as "W_commerce" (in the middle of the rank) which are likely to be similar to some extent to the target domain. KL instead produces a more consistent ranking, where all the spoken domains are lower than the written ones and intuitively similar domains such as "W_commerce" and "W_social_science" are ranked highest. One possibility is that the difference is due to the fact that

W_world_affairs				
R	Domain	KL	Domain	X^2
1	W_world_affairs	0	W_world_affairs	0
2	W_soc_science	0.6770	S_demog_unclassified	1,363,840
3	W_commerce	0.7449	S_cg_public_instit	1,568,540
4	W_arts	0.8205	S_cg_education	1,726,960
5	W_leisure	0.8333	W_belief_thought	1,765,690
6	W_belief_thought	1.0405	S_cg_leisure	1,818,110
7	W_app_science	1.0685	S_cg_business	1,882,430
8	W_nat_science	1.4683	S_demog_DE	2,213,530
9	W_imaginative	1.4986	W_commerce	2,566,750
10	S_cg_education	1.5010	W_arts	2,666,730
1	S_cg_public_instit	1.6694	S_demog_C1	2,668,690
12	S_cg_leisure	1.7632	S_demog_C2	2,716,090
13	S_cg_business	1.8945	S_demog_AB	2,834,220
14	S_demog_AB	2.6038	W_soc_science	3,080,840
15	S_demog_C1	2.7667	W_leisure	3,408,090
16	S_demog_C2	2.8110	W_nat_science	3,558,870
17	S_demog_DE	3.2886	W_app_science	3,711,010
18	S_demog_unclassified	4.3921	W_imaginative	5,819,810

Table 3. Rankings produced by KL and X^2 with respect to the domain "W_world_affairs"

the unigram distributions compared with KL are smoothed while raw counts are used for X^2. However, when we tried smoothing the contingency tables for X^2 we obtained even more inconsistent results. An alternative explanation relates the behavior of X^2 to the fact that the distributions being compared have long tails of low frequency counts. It is a matter of contention whether X^2, in the presence of sparse data, i.e., in the presence of cells with less than five counts, produces results which are appropriately approximated by the χ^2 distribution, and thus statistically interpretable (cf. Agresti 1990). It might be that, even if the use described here only aims at relative assessments of dependency/similarity, rather than parametric testing, the presence of large numbers of low frequency counts causes more noisy measurements with X^2 than with KL.

Different metrics have different properties and might provide different advantages and shortcomings depending on the specific task. Since it seems that KL is more appropriate to our task in the remainder of the paper we mainly present results using KL, although we did run all experiments with both measures, often obtaining very similar results.

3.4 A ranking function for sampled unigram distributions

What properties distinguish unigram distributions drawn from the whole BNC from distributions drawn from its subsets – genre, mode and domain? This is an important question because, if identified, such properties might help to discriminate between sampling methods which produce more random collections of documents and more biased ones. We suggest the following hypothesis. Unigrams sampled from the full BNC have distances from biased samples which tend to be lower than the distances of biased samples to other biased samples. If this hypothesis is true then if we sample unigrams from the whole BNC, and from its "biased" subsets, the

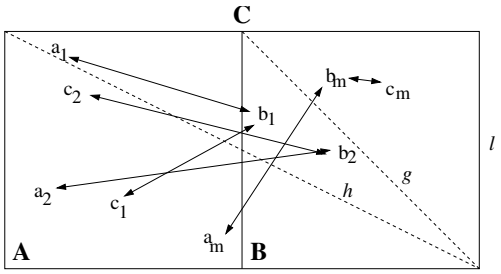

Figure 2. Visualization of the distances (continuous lines with arrows) between points representing unigrams distributions, sampled from "biased" partitions A and B and from the full collection of documents $C = A \cup B$

vector of distances between the BNC sample and all other samples should have lower mean than the vectors for biased samples.

Figure 2 depicts a geometric interpretation of the intuition behind this hypothesis. Suppose that the two squares A and B represent two partitions of the space of documents C. Additionally, m pairs of unigram distributions, represented as points, are produced by random samples of documents from these partitions; e.g., a_1 and b_1. The mean Euclidean distance between (a_i, b_i) pairs is a value between 0 and h, the length of the diagonal of the rectangle which is the union of A and B. Instead of drawing pairs we can draw triples of points, one point from A, one from B, and another point from $C = A \cup B$. Approximately half of the points drawn from C will lie in the A square, while the other half will lie in the B square. The distance of the points drawn from C from the points drawn from B will be between 0 and g, for approximately half of the points (those lying in the B region), while the distance is between 0 and h for the other half of the points (those in A). Therefore, if m is large enough, the average distance between C and B (or A) must be smaller than the average distance between A and B.[2]

[2]Because $h = \sqrt{l^2 + 2l^2} > g = \sqrt{2l^2}$.

Samples from biased portions of the corpus should tend to "remain" in a given region, while samples from the whole corpus should be closer to biased samples, because the unbiased sample draws words from across the whole vocabulary, while biased samples have access to a limited vocabulary. To summarize then, we suggest the hypothesis that samples from the full distribution have a smaller mean distance than all other samples. More precisely, let $U_{i,k}$ be the kth of N unigram distributions sampled under y_i, $y_i \in Y$, where Y is the set of sampling categories. Additionally, for clarity, we will always denote with y_1 the unbiased sample, while y_j, $j = 2..|Y|$, denote the biased samples. Let $\mathbf{M}$ be a matrix of measurements, $\mathbf{M} \in \mathbb{R}^{|Y|\times|Y|}$, such that $M_{i.j} = \frac{\sum_{k=1}^{N} D(U_{i,k},U_{j,k})}{N}$, where $D(.,.)$ can be any similarity measure of the kind discussed above, i.e., X^2 or KL. In other words, the matrix contains the average distances between pairs of samples (biased or unbiased). Each row $M_i \in \mathbb{R}^{|Y|}$ contains the average distances between y_i and all other ys, including y_i. We assign a score δ_i to each y_i which is equal to the mean of the vector M_i (excluding $M_{i,j}$, $j = i$):

$$\delta_i = \frac{1}{|Y| - 1} \sum_{j=1, j\neq i}^{|Y|} M_{i,j} \tag{4}$$

It could be argued that also the variance of the distances for y_1 should be lower than the variance of the other ys, because the unbiased sample tends to be equidistant from all other samples. We will show empirically that this seems in fact to be the case. When the variance is used in place of the mean, δ_i is computed as the traditional variance of M_i (excluding $M_{i,j}$, $j = i$):

$$\delta_i = \frac{1}{|Y| - 2} \sum_{j=1, j\neq i}^{|Y|} [M_{i,j} - \mu_i]^2 \tag{5}$$

where μ_i is the mean of M_i, computed as in equation (4).

3.5 Randomness of BNC samples

We first tested our hypothesis on the BNC in the following way. For each of the three main partitions, mode, domain, and genre, we sampled with replacement (from a distribution determined by relative frequency in the relevant set) 1,000 words from the whole BNC and from each of the labels (categories) belonging to the specific partitions. Then we measured the average distance between each label in a partition, plus the sample from the whole BNC. We repeated this experiment 100 times and summarized the results by ranking each label, within each partition type, using δ.

Table 4 summarizes the results of this experiment for all three partitions: mode, domain, and genre (only partial results are shown for genre). The table shows results obtained both with KL and X^2 to illustrate the kinds of problems mentioned above concerning X^2, but we will focus mainly on the results concerning KL. For all three experiments each sample category y_i is ranked according to its score δ_i. The KL-based δ always ranks the unbiased sample "BNC_all" higher than all other categories. At the top of the rankings we also find other less narrowly topic/genre-dependent categories such as "W" (all written texts) for mode, or "W_misc" and "W_pop_lore" for genre. Thus, our hypothesis is supported by these experimental results. Unbiased samples tend to be closer on average to biased samples, and this property can be used to distinguish a biased from an unbiased unigram sampling method. Interestingly, as anticipated in section 3.4, also the variance of the distance vector seems to correlate well with "biased-ness". Unbiased samples tend to have more constant distances from biased samples, than samples to one another. Table 5 summarizes the – comparable – results obtained using for δ_i equation (5); e.g., the variance of M_i.

A different story holds for X^2. There is clearly something wrong in the rankings, although, sometimes we find the unbiased sample ranked the highest. For example, for mode, "S" (spoken) is

Rankings, based on δ-mean								
	Mode		Domain			Genre		
	X^2	KL	X^2	KL		X^2	KL	
R								
1	BNC_all	BNC_all	S_cg_business	BNC_all		S_meeting	BNC_all	
2	S	W	S_demog_C1	S_cg_education		S_speech_unscripted	W_misc	
3	W	S	S_demog_C2	W_leisure		S_brdcast_discussion	W_pop_lore	
4			S_demog_AB	W_arts		S_interview	W_non_ac_soc_sci	
5			S_cg_leisure	W_belief_thought		S_unclassified	W_non_ac_humanities_arts	
6			S_demog_DE	W_imaginative		S_tutorial	W_newsp_brdsht_nat_misc	
7			S_cg_education	S_cg_leisure		S_interview_oral_hist	W_newsp_other_soc	
8			S_cg_public_inst	S_cg_business		S_courtroom	W_biography	
9			S_demog_unclass	W_app_sci		S_lect_humanities_arts	W_non_ac_nat_sci	
10			BNC_all	W_soc_sci		S_brdcast_documentary	W_ac_humanities_arts	
11			W_imaginative	S_cg_public_inst		S_lect_soc_sci	W_newsp_other_report	
12			no_cat	W_world_affairs		S_parliament	W_newsp_brdsht_nat_arts	
13			W_belief	W_commerce		S_brdcast_news	W_newsp_brdsht_nat_soc	
14			W_soc_sci	W_nat_sci		S_lect_polit_law_edu	S_brdcast_news	
15			W_commerce	S_demog_AB		S_classroom	S_brdcast_discussion	
16			W_leisure	S_demog_C1		S_consult	W_newsp_tabloid	
17			W_arts	S_demog_C2		S_pub_debate	W_newsp_other_arts	
18			W_app_sci	S_demog_DE		S_conv	W_newsp_brdsht_nat_edit	
19			W_world_affairs	S_demog_unclass		S_speech_scripted	W_newsp_other_sci	
20			W_nat_sci	no_cat		S_sermon	W_newsp_brdsht_nat_report	
21						S_demonstration	W_advert	
22						W_non_ac_soc_sci	W_ac_soc_sci	
23						BNC_all	W_commerce	
..						..	..	
68						W_fict_drama	S_sportslive	
69						W_non_ac_tech_engin	S_consult	
70						W_ac_medicine	W_fict_drama	
71						W_ac_nat_sci	S_lect_commerce	
72						W_fict_poetry	no_cat	

Table 4. Rankings based on δ, as the mean distance between samples from the BNC partitions plus samples from the whole BNC; low values for δ ranked higher

Rankings based on δ-variance								
	Mode		Domain		Genre			
R	X^2	KL	X^2	KL	X^2	KL		
1	BNC_all	BNC_all	S_cg_public_inst	BNC_all	BNC_all	BNC_all		
2	S	W	S_cg_business	W_leisure	W_misc	W_pop_lore		
3	W	S	S_cg_education	W_arts	W_non_ac_soc_sci	W_misc		
4			BNC_all	W_imaginative	W_non_ac_med	S_brdcast_news		
5			S_cg_leisure	W_belief_thought	W_newsp_other_sci	W_non_ac_nat_sci		
6			W_belief_thought	S_cg_education	S_brdcast_news	W_non_ac_soc_sci		
7			W_imaginative	W_app_sci	W_pop_lore	W_newsp_brdsht_nat_arts		
8			W_arts	S_cg_public_inst	W_newsp_brdsht_nat_soc	W_non_ac_humanities_arts		
9			no_cat	W_world_affairs	W_newsp_brdsht_nat_sci	W_biography		
10			W_leisure	W_soc_sci	S_brdcast_documentary	W_ac_humanities_arts		
11			W_soc_sci	W_commerce	W_letters_personal	W_newsp_brdsht_nat_misc		
12			W_commerce	W_nat_sci	W_newsp_brdsht_nat_edit	W_newsp_other_soc		
13			W_world_affairs	S_cg_business	W_non_ac_humanities_arts	W_essay_school		
14			W_app_sci	S_cg_leisure	W_newsp_other_soc	W_fict_prose		
15			W_nat_sci	S_demog_unclas	W_biography	W_newsp_brdsht_nat_sci		
16			S_demog_unclas	S_demog_AB	W_religion	W_newsp_brdsht_nat_soc		
17			S_demog_AB	S_demog_C2	W_essay_school	W_non_ac_med		
18			S_demog_C1	S_demog_C1	W_newsp_brdsht_nat_misc	W_fict_poetry		
19			S_demog_C2	S_demog_DE	W_non_ac_nat_sci	W_advert		
20			S_demog_DE	no_cat	W_essay_univ	W_religion		
.					..	..		
68					S_interview	S_unclassified		
69					S_unclassified	S_lect_commerce		
70					S_conv	no_cat		
71					S_classroom	S_classroom		
72					S_consult	S_consult		

Table 5. Rankings based on δ, as the variance of the average distance between samples from the BNC partitions plus samples from the whole BNC; low values for δ ranked higher

ranked higher than "W", but it seems counterintuitive that samples from only 5% of all documents are on average closer to all samples than samples from 95% of documents. The reason why in general "S" categories tend to be closer (also in the domain and genre experiments) might have to do with low counts as suggested before, and it may also be related to the magnitude of the unigram lists; i.e., distributions made of a small number of unigrams might tend to be closer to other distributions because of the small number of words involved independently of the actual "similarity".

4 Evaluating the randomness of corpora derived from Google

In our proof-of-concept experiment, we compared the distribution of words drawn from the whole BNC to those of words that belong to various categories. Of course, when we download documents from the Web via a search engine (or sample them in other ways), we cannot choose to sample random documents from the whole Web, nor select documents belonging to a certain category. We can only use specific lexical forms as query terms, and we can only retrieve a fixed maximum number of pages per query. Moreover, while we can be relatively confident that the retrieved pages will contain all the words in the query, we do not know according to which criteria the search engine selects the pages to return among the ones that match the query.[3] All we can do is to try to control the typology of documents returned by using specific query terms (or other means), and we can use a measure such as the one we proposed to look for the least biased retrieved collection among a set of retrieved collections.

[3] If not in very general terms, e.g., it is well known that Google's PageRank algorithm weights documents by popularity.

4.1 Selection of query terms

Since the query options of a search engine do not give us control over the genre, topic and other textual parameters of the documents to be retrieved, we must try to construct a "balanced" corpus by selecting appropriately balanced query terms, e.g., using random terms extracted from an available balanced corpus (see Sharoff this volume). In order to build specialized domain corpora, we will have to use "biased" query terms from the appropriate domain (see Baroni and Bernardini 2004). We extract the random terms from the clean, balanced, 1M-words Brown corpus of American English (Kučera and Francis 1967). Since the Web is likely to contain much larger portions of American than British English, we felt that queries extracted from the BNC would be overall more biased than American English queries. We extracted the top 200 most frequent words from the Brown ("high frequency" set), 200 random terms with frequency between 100 and 50 inclusive ("medium frequency" set) and 200 random terms with minimum frequency 10 (the "all frequency" set – because of the Zipfian properties of word types, this is a *de facto* low frequency word set). We experimented with each of these lists as ways to retrieve an unbiased set of documents from Google. Notice that there are arguments for each of these selection strategies as plausible ways to get an unbiased sample from the search engine: high frequency words are not linked to any specific domain; medium and low frequency words sampled randomly from a balanced corpus should be spread across a variety of domains and styles.

In order to build biased queries, that should hopefully lead to the retrieval of sets of topically related documents, we randomly extracted lists of 200 words belonging to the following 10 domains from the topic-annotated extension (Magnini and Cavaglia, 2000) of WordNet (Fellbaum, 1998): *administration, commerce, computer science, fashion, gastronomy, geography, military, music, sociology.* These domains were chosen since they look "general"

enough to be very well-represented on the Web, but not so general as to be virtually unbiased (cf. the WordNet domain *person*). We selected words only among those that did not belong to more than one WordNet domain, and we avoided multi-word terms.

4.2 Experimental setting

From each source list ("high", "medium" and "all" frequency sets plus the 10 domain-specific lists), we randomly select 20 pairs of words without replacement (i.e., no word among the 40 used to form the pairs is repeated). We use each pair as a query to Google, asking for pages in English only (we use pairs instead of single words to maximize our chances to find documents that contain running text – see discussion in Sharoff this volume). For each query, we retrieve a maximum of 20 documents. The whole procedure is repeated 20 times with all lists, so that we can compute means and variances for the various quantities we calculate.

Our unit of analysis is the corpus constructed by putting together all the non-duplicated documents retrieved with a set of 20 paired word queries. However, the documents retrieved from the Web have to undergo considerable post-processing before being usable as parts of a corpus. In particular, following what is becoming standard practice in Web corpus construction (see, e.g., Fletcher 2004), we discard very large and very small documents (documents larger than 200Kb and smaller than 5Kb, respectively), since they tend to be devoid of linguistic content and, in the case of large documents, can skew the frequency statistics. For technical reasons, we focus on HTML documents, discarding, e.g., PDF files. Moreover, we use a re-implementation of the heuristic used by Aidan Finn's BTE tool[4] to identify and extract stretches of connected prose and discard "boilerplate". In short, the method looks for and selects the fragment of text where the difference between

[4] `http://smi.ucd.ie/hyppia/bte/`

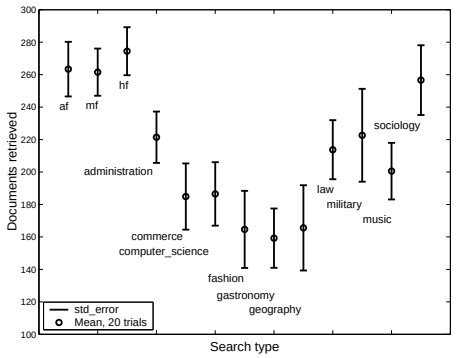

Figure 3. Average number of documents retrieved for each query category over the 20 search sets; the error bar represents the standard deviation

text token count and HTML tag count is maximal. As a further filter, we only keep documents where at least 25% of the tokens in the stretch of text extracted in the previous step are from the list of 200 most frequent Brown corpus words. Because of the Zipfian properties of texts, it is pretty safe to assume that almost any well-formed stretch of English connected prose will satisfy this constraint.

Notice that a corpus can contain maximally 400 documents (20 queries times 20 documents retrieved per query), although typically the documents retrieved are not as many, because different queries retrieve the same documents, or because some query pairs are found in less than 20 documents. Figure 3 plots the means (calculated across the 20 repetitions) of the number of documents retrieved for each query category, and table 6 reports the sizes in types and tokens of the resulting corpora. Queries for the "unbiased" seeds (af, mf, and hf) tend to retrieve more documents, although most of the differences are not statistically significant and, as the table shows, the difference in number of documents is often counterbalanced by the fact that specialized queries tend to retrieve longer documents. The difference in number of doc-

Search category	Avg types	Avg tokens
af	35,988	441,516
mf	32,828	385,375
hf	39,397	477,234
administration	39,885	545,128
commerce	38,904	464,589
computer_science	25,842	311,503
fashion	44,592	533,729
gastronomy	36,535	421,705
geography	42,715	498,029
law	49,207	745,434
military	47,100	667,881
music	45,514	558,725
sociology	56,095	959,745

Table 6. Average number of types and tokens in corpora constructed via Google queries

uments retrieved does not seem to have any systematic effect on the resulting distances, as will be briefly discussed in 4.5 below.

4.3 Distance matrices and bootstrap error estimation

We now rank each individual query category y_i, biased and unbiased, using δ_i, as we did before using the BNC partitions (cf. section 3.5). Unigram distributions resulting from different search strategies are compared by building a matrix of mean distances between pairs of unigram distributions. Rows and columns of the matrices are indexed by the query category, the first category corresponds to one unbiased query, while the remaining indexes correspond to the biased query categories; i.e., $M \in \mathbb{R}^{11\times 11}$, $M_{i,j} = \frac{\sum_{k=1}^{20} D(U_{i,k},U_{j,k})}{20}$, where $U_{s,k}$ is the kth unigram distribution produced with query category y_s.

The data collected can be seen as a dataset $\mathcal{D}$ of $n = 20$ datapoints each consisting of a series of unigram word distributions,

one for each search category. If all n data-points are used once to build the distance matrix we obtain one such matrix for each unbiased category. Based on such matrix we can rank a search strategy y_i using δ_i as explained above (cf. section 3.4). Instead of using all n data-points once, we create B "bootstrap" datasets (cf. Duda et al. 2001) by randomly selecting n data-points from $\mathcal{D}$ with replacement (we used a value of B=100). The B bootstrap datasets are treated as independent sets and they are used to produce B individual matrices M_b from which we compute the score $\delta_{i,b}$, i.e., the mean distance of a category y_i with respect to all other query categories in that specific bootstrap dataset. The bootstrap estimate of δ_i is the mean of the B estimates on the individual datasets:

$$\hat{\delta}_i = \frac{1}{B}\sum_{b=1}^{B}\hat{\delta}_{i,b} \tag{6}$$

Bootstrap estimation can be used to estimate the variance of our measurements of δ_i, and thus the standard error:[5]

$$\sigma_{boot}[\hat{\delta}_i] = \sqrt{\frac{1}{B}\sum_{b=1}^{B}[\hat{\delta}_i - \hat{\delta}_{i,b}]^2} \tag{7}$$

As before we smooth the word counts when using KL, by adding a count of 1 to all words in the overall dictionary. This dictionary is approximated with the set of all words occurring in the unigrams involved in a given experiment, overall on average approximately 1.8 million types (notice that numbers and other special tokens are boosting up this total). Words with an overall frequency greater than 50,000 are treated as stop words and excluded from consideration (188 types).

[5] If the statistic δ is the mean, then in the limit of B the bootstrap estimate of the variance is the variance of δ.

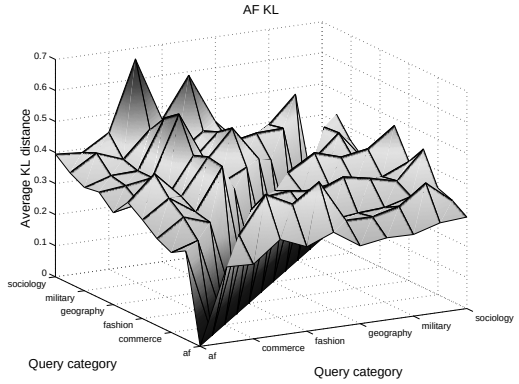

Figure 4. 3D plot of the KL distance matrix comprised of the unbiased query (af) and the biased queries results; only a subset of the biased query labels are shown

4.4 Results

As an example of the kind of results we obtain, figure 4 plots the matrix produced by comparing the frequency lists from all 10 biased queries and the query based on the "all frequency" (af) term set with KL. As expected the diagonal of the matrix contains all zeros, while the matrix is not symmetric. The important thing to notice is the difference between the vectors regarding the unbiased query; i.e., $M_{1,j}$ and $M_{i,1}$ and the other vectors. The unbiased vectors are characterized by smaller distances than the other vectors. They also have a "flatter", or more uniform, shape. The experiments involving the other unbiased query types, "medium frequency" and "high frequency", produce similar results.

The upper half of table 7 summarizes the results of the experiments with Google, compiled by using the mean KL distance. The unbiased sample (af, mf, and hf) is always ranked higher than all biased samples. Notice that the bootstrapped error estimate shows that the unbiased sample is significantly more random than the others. Interestingly, as the lower half of table 7 shows, somewhat similar results are obtained using the variance of the vectors

Rankings with Bootstrap error estimation, $\hat{\delta}$ = mean distance									
R	Sample	$\hat{\delta}_i$	$\sigma_{boot}[\hat{\delta}_i]$	Sample	$\hat{\delta}_i$	$\sigma_{boot}[\hat{\delta}_i]$	Sample	$\hat{\delta}_i$	$\sigma_{boot}[\hat{\delta}_i]$
1	af	0.13040	0.001892	mf	0.12470	0.002176	hf	0.13082	0.002368
2	commerce	0.14997	0.007186	commerce	0.15062	0.007273	commerce	0.14989	0.007177
3	geography	0.16859	0.009102	geography	0.16986	0.009061	geography	0.16907	0.009152
4	admin	0.17254	0.004040	admin	0.17338	0.004081	admin	0.17257	0.004035
5	fashion	0.17292	0.007944	fashion	0.17403	0.007981	fashion	0.17313	0.007893
6	comp_sci	0.17437	0.004554	comp_sci	0.17486	0.004651	comp_sci	0.17408	0.004605
7	military	0.19181	0.007113	military	0.19388	0.007160	military	0.19192	0.006976
8	gastronomy	0.19560	0.009307	gastronomy	0.19708	0.009461	music	0.19612	0.006689
9	music	0.19583	0.006611	music	0.19761	0.006754	law	0.19635	0.005669
10	law	0.19707	0.005661	law	0.19900	0.005718	gastronomy	0.19646	0.009335
11	sociology	0.24075	0.008674	sociology	0.24329	0.008596	sociology	0.24023	0.008496
Rankings with Bootstrap error estimation, $\hat{\delta}$ = variance									
R	Sample	$\hat{\delta}_i$	$\sigma_{boot}[\hat{\delta}_i]$	Sample	$\hat{\delta}_i$	$\sigma_{boot}[\hat{\delta}_i]$	Sample	$\hat{\delta}_i$	$\sigma_{boot}[\hat{\delta}_i]$
1	af	0.00018	3.06478e-05	mf	0.00023	3.03428e-05	hf	0.00019	3.07505e-05
2	music	0.00028	2.52644e-05	music	0.00026	2.26974e-05	music	0.00028	2.47110e-05
3	commerce	0.00029	4.45361e-05	commerce	0.00031	4.10216e-05	commerce	0.00030	4.39917e-05
4	fashion	0.00043	6.83792e-05	fashion	0.00043	6.94822e-05	fashion	0.00043	7.07811e-05
5	geography	0.00046	6.43744e-05	geography	0.00046	6.61234e-05	geography	0.00044	6.74963e-05
6	gastronomy	0.00066	7.31346e-05	gastronomy	0.00065	6.90454e-05	gastronomy	0.00062	6.39204e-05
7	comp_sci	0.00068	5.57851e-05	comp_sci	0.00075	6.14489e-05	comp_sci	0.00072	5.73490e-05
8	admin	0.00079	8.58979e-05	admin	0.00082	8.32991e-05	admin	0.00081	8.86421e-05
9	military	0.00094	0.000114039	military	0.00091	0.000116997	military	0.00095	0.000120596
10	law	0.00147	0.000145864	law	0.00145	0.000152849	law	0.00154	0.000152587
11	sociology	0.00296	0.000295807	sociology	0.00293	0.000307310	sociology	0.00302	0.000323409

Table 7. Google experiments: rankings for each unbiased sample category with bootstrap error estimation (B=100)

M_i instead of the mean, to compute δ_i. The unbiased method is always ranked highest. However, since the specific rankings produced by mean and variance show some degree of disagreement, it is possible that a more accurate measure could be obtained by combining the two measures.

4.5 Discussion

We observed, on Google, the same behavior that we saw in the BNC experiments, where we could directly sample from the whole unbiased collection and from biased subsets of it (documents partitioned by mode, domain and genre). This provides support for the hypothesis that our measure can be used to evaluate how unbiased a corpus is, and that issuing unbiased/biased queries to a search engine is a viable, nearly knowledge-free way to create unbiased corpora, and biased corpora to compare them against.

If our measure is quantifying unbiased-ness, then the lower the value of δ with respect to a fixed set of biased samples, the better the corresponding seed set should be for the purposes of unbiased corpus construction. In this perspective, our experiments also show that unbiased queries derived from "medium frequency" terms (e.g., *places*, *wonderful*) perform better than all frequency (therefore mostly low frequency) and high frequency terms (e.g., *soils*, *contraction* and *even*, *what*, respectively). Thus, while more testing is needed, our data provide some support for the choice of words that are neither too frequent nor too rare as seeds, when building a Web-derived corpus.

Finally, the results indicate that, despite the fact that different query sets retrieve on average different amounts of documents, and lead to the construction of corpora of different lengths, there is no sign that these differences are affecting our δ measure in a systematic way; e.g., some of the larger collections, in terms of number of documents and token size, are both at the top (the unbiased samples) and at the bottom of the ranks (law, sociology)

in table 7.

5 Conclusion

As research based on the Web as corpus, and in particular on automated Web-based corpus construction, becomes more prominent within computational and corpus-based linguistics, many fundamental issues have to be tackled in a more systematic way. Among these, there is the problem of assessing the quality and nature of a corpus built with automated means.

In this paper, we considered one particular approach to automated corpus construction (via search engine queries for combinations of a set of seed words), and we proposed an automated, quantitative, nearly knowledge-free way to evaluate how "biased" a corpus constructed in this way is. Our method is based on the idea that the frequency distribution of words in an unbiased collection will be, on average, less distant from distributions derived from biased partitions, than any of the biased distributions (we showed that this is indeed the case for a collection where we have access to the full unbiased and biased distributions, i.e., the BNC), and on the idea that biased collections of Web documents can be created by issuing "biased" queries to a search engine.

The results of our experiments with Google, besides confirming the hypothesis that corpora created using unbiased seeds have lower average distance to corpora created using biased seeds, compared to the average distance of each biased corpus to the others biased corpora, suggest that the seeds to build an unbiased corpus should be selected among medium frequency words (medium frequency in an existing balanced corpus, that is), rather than among high frequency words or words not weighted by frequency (as in the setting in which we sampled from the whole Brown type list).

We realize that our study leaves many questions open, each of them corresponding to an avenue for further study. One of the

crucial issues is what it means for a corpus to be unbiased. As we already stressed, we do not necessarily want our corpus to be an unbiased sample of what is out there on the Net – we want it to be composed of content-rich pages, and reasonably balanced in terms of topics and genres, despite the fact that the Web is unlikely to be balanced in terms of topics and genres. Issues of representativeness and balance of corpora are widely discussed by corpus linguists (see Kilgarriff and Grefenstette 2003 for an interesting perspective on these issues from the point of view of Web-based corpus work). For our purposes, we implicitly define balance in terms of the set of biased corpora that we compare the target corpus against. Assuming that our measure of unbiased-ness/balance is appropriate, all it tells us is that a certain corpus is more/less biased than another corpus with respect to the biased corpora we compared them against (e.g., in our case, the corpus built with mid frequency seeds is less biased than the others with respect to corpora that represent 10 broad topic-based WordNet categories). Thus, it will be important to check whether our methodology is stable across choices of biased samples. In order to verify this, we plan to replicate our experiments using a much higher number of biased categories, and systematically varying the biased categories. We believe that this should be made possible by sampling biased documents from the long lists of pre-categorized pages in the Open Directory Project (`http://dmoz.org/`).

Our WordNet-based queries are obviously aimed at creating corpora that are biased in terms of *topics*, rather than *genres* or *textual types*. A balanced corpus should also be unbiased in terms of genres. In order to apply our method to genre-based balancing, we need to devise ways of constructing corpora that are genre-specific, rather than topic-specific. This is a more difficult task, not least because the whole notion of what exactly is a "Web genre" is far from settled (see, e.g., Santini 2005). Moreover, while sets of seed words can be used to retrieve words belonging to a certain

topic, it is less clear how genres can be targeted through search engine queries. Again, the Open Directory Project categorization could be helpful here, as it seems to be, at least in part, genre-based (e.g., the Science section is organized by topic – agriculture, biology, etc. – but also into categories that are likely to correlate, at least partially, with textual types: chats and forums, educational resources, news and media, etc.)

We tested our method on three rather similar ways to select unbiased seeds (all based on the extraction of words from an existing balanced corpus). Corpora created with seeds of different kinds (e.g., basic vocabulary lists, as in Ueyama this volume) should also be evaluated. Indeed, a long term goal would be to use our method to iteratively bootstrap "optimal" seeds, starting from an arbitrary seed set. More in general, the method is not limited to the evaluation of corpora built via search engine queries. For example, it would be interesting to compare the randomness of corpora built in this way to that of corpora built by Web crawls that start from a set of seed URLs (e.g., Emerson and O'Neil this volume).

Finally, we would like to explore extensions of our method that could be applied to the analysis of corpora in general (Web-derived or not), both for the purpose of evaluating their relative degree of biased-ness, and as a general-purpose corpus comparison technique (on corpus comparison, see, e.g., Kilgarriff (2001).

References

Agresti, A. (1990). *Categorical data analysis*, New York: Wiley.

Aston, G. and Burnard, L. (1998). *The BNC handbook: Exploring the British National Corpus with SARA*, Edinburgh: Edinburgh University Press.

Bar-Yossef, Z., Berg, A., Chien, S., Fakcharoenphol, J. and Weitz,

D. (2000). Approximating aggregate queries about Web pages via random walks. *Proceedings of VLDB 2000*, 535-544.

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. *Proceedings of LREC 2004*, 1313-1316.

Bharat, K. and Broder, A. (1998). A technique for measuring the relative size and overlap of the public Web search engines. *Proceedings of WWW7*, 379-388.

Biber, D. (1993). Representativeness in corpus design. *Literary and Linguistic Computing* 8, 1-15.

Cover, T.M. and Thomas, J.A. (1991). *Elements of information theory*, New York: Wiley.

Duda, R.O., Hart, P.E. and Stork, D.G. (2001). *Pattern classification, 2nd ed.*, New York: Wiley.

Fellbaum, C. (ed.) (1998). *WordNet: An electronic lexical database*, Cambridge: MIT Press.

Fletcher, B. (2004). Making the Web more useful as a source for linguistic corpora. In Connor, U. and Upton, T. (eds.) *Corpus linguistics in North America 2002*, Amsterdam: Rodopi.

Ghani, R., Jones, R. and Mladenić, D. (2001). Mining the Web to create minority language corpora. *Proceedings of the 10th International Conference on Information and Knowledge Management*, 279-286.

Kilgarriff, A. (2001). Comparing corpora. *International Journal of Corpus Linguistics* 6, 1-37.

Kilgarriff, A. and Grefenstette, G. (2003). Introduction to the special issue on the Web as corpus. *Computational Linguistics* 29(3), 333-347.

Henzinger, M., Heydon, A. and Najork, M. (2000). On near-uniform URL sampling. *Proceedings of WWW9*, 295-308.

Kučera, H. and Francis, W.N. (1967). *Computational analysis of present-day American English*, Providence: Brown University Press.

Lee, D. (2001). Genres, registers, text types, domains and styles: Clarifying the concepts and navigating a path through the BNC jungle. *Language Learning & Technology* 5(3), 37-72.

Magnini, B. and Cavaglia, G. (2000). Integrating subject field codes into WordNet. *Proceedings of LREC 2000*, 1413-1418.

Rayson, P. and Garside, R. (2000). Comparing corpora using frequency profiling. *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*, 1-6.

Santini, M. (2005). Genres in formation? An exploratory study of Web pages using cluster analysis. *Proceedings of the 8th Annual Colloquium for the UK Special Interest Group for Computational Linguistics*.

Shannon, C.E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27.

Sharoff, S. (Submitted). Open-source corpora: Using the Net to fish for linguistic data.

Ueyama, M. and Baroni, M. (2005). Automated construction and evaluation of a Japanese Web-based reference corpus. *Proceedings of Corpus Linguistics 2005*, available online at `http://www.corpus.bham.ac.uk/PCLC/`.

Using the Web as a Source of LSP Corpora in the Terminology Classroom

Sara Castagnoli

1 A short introduction to corpus-based terminology

Corpus-based terminology can be described as a working method which consists in exploring a domain-specific corpus in order to investigate terminological issues (Gamper and Stock 1998).

Even though its theoretical grounds are similar to those on which corpus-based lexicography is founded, it has taken longer for corpus-based terminology to become an established procedure; this is probably due to the different nature of the corpora involved, which are large and general – and therefore easily reusable – in the former case, domain-specific and smaller – i.e., difficult to re-use – in the latter. Terminologists and translators usually need to build a new corpus every time they embark on a new task, and the consequent reduced cost-effectiveness has often been adduced as the main argument against the construction of "disposable" (as defined in Varantola 2003) corpora, especially in relation to those domains in which most reference material used to be available only on paper, thus requiring manual checking or scanning. Today, however, the increased availability of texts in electronic format enables to speed up the process of collecting and processing corpora to an extent which was unthinkable until not so long ago.

2 And here comes the Web...

Being an unparalleled, virtually unlimited and ever expanding source of machine-readable texts, encompassing almost every language and knowledge domain (Fletcher 2004), the Web can play a leading role for the use of corpora to become common practice both in translation and terminology. While we do not believe that the Web can be considered a corpus – and certainly not a specialized corpus – in itself, since its contents are not assembled according to any specific criteria, we will argue that it may represent a good source for LSP (language for special purposes) corpora and terminology, for a variety of reasons.

First of all, as mentioned above, it is possible to find on the Internet texts on virtually any specialized subject, written in a variety of genres and communicative settings (expert-expert, expert-initiated/uninitiated[1] and, even if less interesting for the purposes of terminological research, initiated-initiated/uninitiated), which allows terminologists to choose among sources characterized by different levels of specialization, and to study variation and synonymy across different text types.

Secondly, while being a drawback for other types of linguistic research, the fact that new documents appear or are updated on the Web on a daily basis is an asset for terminologists: since terms are continually being invented and evolving, in relation to both their meaning and usage, it can be argued that a Web-based open corpus is more likely to contain up-to-date terms and state-of-the-art concepts than a static corpus.

Lastly, besides the fact that Web access is becoming increas-

[1] Pearson (1998) describes how specialized terms can occur in different communicative settings, arguing that terminological density varies according to the degree of specialization of the participants. "Initiates" are defined as people having some knowledge of a given specialized field, whereas "uninitiated (...) are not necessarily involved, either professionally or through their leisure interests, in a particular subject field".

ingly easier and inexpensive, and that it is constantly available, most translators are already familiar with it and use it in their everyday work,[2] which makes it reasonable to suggest that tools for corpus creation and analysis based on the Web would be easily integrated into their workstations.

3 Teaching corpus-based terminology using the Web

Given that terminological research constitutes a substantial part of the translator's work, and that corpora – both general and specialized – have been suggested to be effective tools in enhancing the quality of translations (Gavioli and Zanettin 1997), the principles and methodology involved in creating corpora and extracting terminology from them have become part of the teaching curriculum at the School for Interpreters and Translators of the University of Bologna, Forlì, Italy.

This paper reports on a classroom experience carried out in Spring 2005 with a group of ten trainee translators taking an optional 48-hour course in Terminology and LSP. The main objective of the course was to teach students why and how to use corpora in two stages of terminology work, namely term extraction and terminography (i.e., the recording and presentation of terminological data, most often by means of databases). The course was mainly organized along these two axes, developed by two different teach-

[2] A questionnaire circulated to professional translators during the period April-June 2005 in the framework of the European project MeLLANGE (*Multilingual e-Learning in LANGuage Engineering*, `http://mellange.eila.jussieu.fr/`) revealed that – over 623 respondents, located mainly in the UK, but also in France, Italy and Germany – 93.4% of translators use Google to research terminology, with more or less refined strategies, 43.3% regularly visit websites belonging to specific companies, 29.6% regularly visit websites acting as domain portals and 21% regularly visit other kinds of websites.

ers;[3] the module on corpus creation – preceded by a few lessons on the Unix operating system – was integrated with introductory notes on corpus annotation, XML, POS tagging and collocation extraction, whereas the final lessons were dedicated to illustrate the use of termbases within some CAT tools. Our aim was to consider terminological work both as an autonomous discipline and as a component of the translation process. Corpora were thus created and analyzed in two different teaching situations, i.e., during the terminology course proper and for the end-of-course project.

Since corpus creation was not the main subject of the course, and since designing and constructing "well-made" corpora would have required much more time and effort than available in the classroom, students were asked to work on corpora assembled automatically using the BootCaT toolkit, a suite of Perl programs designed to bootstrap specialized corpora from the Web (Baroni and Bernardini 2004). Other reasons behind this choice included the desire to introduce students to a tool which they might find helpful for their future activity as translators, as well as to provide them with new IT competences. Advantages and disadvantages of automatic corpus compilation were then discussed with students on the basis of their analysis of the usefulness of their corpora, which proved to be a very instructive activity. Some of the conclusions that were reached are reported in the following sections.

3.1 During the course: Practicing term extraction

After introducing students to the basic principles of terminology (languages for special purposes vs. general language, terms vs. words, terms vs. concepts, etc.), and having illustrated the advantages of corpora over traditional dictionaries, students were asked to choose domains they were familiar or had already worked

[3] Alessandra Matteucci was in charge of the part of the course about terminography.

with in other translation courses, and to provide a list of terms they presumed to be typical of such domains, to be used as seeds for the Web mining procedure. The aim of the exercise was to collect a corpus on which to practice term extraction through a variety of techniques, such as the production of word or cluster (bi-grams, tri-grams, etc.) lists, statistical measures (frequency, mutual information and log-likelihood), and morphosyntactic analysis (based on POS tagging, i.e., retrieving all occurrences of given combinations of POS tags which are hypothesized to be typical patterns for terms, such as ADJ+NOUN or NOUN+NOUN in English). The reason we asked students to work on domains they were already acquainted with is that we wanted them to be able to judge the results of the above methods, in order to start a discussion on which term extraction techniques they considered to be more profitable.

The three groups decided to work on medicine (nervous system disorders), law (Italian company law) and technology (cell phones), the first two subjects having been dealt with during a translation course and the third being chosen on the spot, as a domain known to all the members of the group. Table 1 shows the terms chosen as domain key-words for the automatic downloading of Web pages.

Students were then allowed to decide the size and number of tuples to be formed as well as the maximum number of URLs they wanted to retrieve for each tuple, while keeping numbers low enough for the retrieval process not to be too long. Table 2 shows that, although students made more or less the same decisions, the final result – i.e., the size and/or quality of their corpora – differed remarkably. In the following paragraphs we will try to identify possible reasons for this phenomenon by analyzing some data taken from the medical corpus and the cell phone corpus.[4]

[4]Interim data about the Company law corpus are not available because students were not required to document and save data about each single stage

medicine	**company law**	**cell phones**
neurotrasmissione	diritto societario	cellulare
sistema nervoso centrale	decimi	scheda SIM
noradrenalina	pubblicità	PIN
dopamina	azioni	PUK
catecolamina	creditore	GSM
sistema nervoso autonomo	riforma	WAP
sostanza nera	registro delle imprese	UMTS
neurone	spa	GPRS
cellula nervosa	amministratore	SMS
sistema dopaminergico	srl	T9
	pignoramento	MMS
	conferimento	videofonino
	regolamento	bluetooth
	partecipazioni	caricabatteria
	società unipersonale	auricolare
	società pluripersonale	batteria al litio
	consiglio di	infrarossi
	amministrazione	videochiamata
		vivavoce
		scrittura intuitiva
		schermo a cristalli liquidi

Table 1. Seeds for the Web mining procedure

As shown in the first two rows of table 2, different choices were made in relation to the number of tuples. Having chosen a limited number of highly specialized terms, the group working on the medical domain decided to form twenty 2-term tuples, in order to avoid specifying search criteria so narrow that they would probably have resulted in a very small corpus. Instances of such tuples include ["sistema nervoso autonomo" "sistema nervoso centrale"] (*autonomic nervous system, central nervous system*), "sistema nervoso centrale" "sostanza nera"] (*central nervous system, substantia nigra*), ["cellula nervosa" "sostanza nera"] (*nerve cell, substantia nigra*), [noradrenalina neurone] (*noradrenaline neuron*). On the other hand, the cell phone group decided to create fifteen 3-term tuples, such as [cellulare videochiamata GPRS] (*mobile phone,*

of the corpus creation process.

Domain	Medicine	Company law	Cell phones
tuple size	2	n.a.	3
tuples	20	n.a.	15
URLs	183	n.a.	138
URLs/tuples	9.15	n.a.	9.2
lines	37,073	34,821	40,749
words	281,015	281,736	160,298
characters	2,010,356	1,931,760	1,120,754
words/URLs	1,535.60	n.a.	1,161.58
ch.s/URLs	10,985.55	n.a.	8,121.41

Table 2. Corpora statistics

video call, GPRS), [SMS videochiamata UMTS] (*text message, video call, UMTS*), [caricabatteria SMS GSM] (*battery charger, text message, GSM*), [WAP bluetooth "scrittura intuitiva"] (*WAP, bluetooth, predictive text*). Both groups decided to retrieve a maximum of 10 URLs for each tuple, with similar URLs/tuples ratios.

Table 2 shows that there is a remarkable difference in size between the medical corpus and the cell phone corpus, which can be only partly explained by the lower number of tuples searched.

Analysis of average words/URLs and characters/URLs ratios actually allow us to state that webpages related to cell phones are much shorter (by 347 words and 1,864 characters, respectively) than those belonging to the medical domain. Inspection of retrieved URLs and further analysis of the cell phone corpus through word lists (e.g., table 3) and concordances suggest that this is due to the kind of webpages that were downloaded, i.e., pages belonging predominantly to commercial sites or to Web portals offering different kinds of cell phone services (downloading of ringtones and wallpapers, comparison of technical specifications, etc.). Normally such websites are not rich in descriptive or informative pages, but rather conceived with a persuasive purpose and therefore stylistically characterized by eye-catching images and lists; this idea

is corroborated by the evident disparity in the number of lines between the two corpora (see table 2).

On the other hand, observation of the most frequent nouns in the two other corpora suggests that these are largely characterized by highly specialized and formal texts.

medicine		company law		cell phones	
995	sistema	1,864	articolo	871	suonerie
661	cellule	1,770	comma	606	Foto
483	parte	1,689	società	474	telefono
471	cervello	1,179	Art	442	colori
374	cellula	1,147	soci	375	Prezzo
372	neuroni	830	capitale	259	Siemens
371	attività	816	decreto	256	dati
369	membrana	774	azioni	254	Band
354	effetti	753	numero	230	acquisto
351	malattia	746	caso	226	credito
333	azione	705	socio	218	tecnologia
327	corpo	680	amministratori	218	band
307	neurone	668	atto	216	Provenienza
307	farmaci	652	societa'	213	Spese
305	recettori	637	diritto	211	servizi

Table 3. 15 most frequent nouns in the three corpora

One of the first conclusions that can be reached is, therefore, that the automatic creation of corpora from the Web for terminological research is more effective and productive for domains which are highly specialized, whereas it is difficult to retrieve specialized texts concerning more popular domains (e.g., cell phones), in relation to which there is an overflow of information on the Web. Specialized terms belonging to such domains (e.g., "*LCD*", "*lithium battery*") have become so common in everyday language (it might be argued that they have gone through a process of "determinologization", i.e., they have lost their specificity to become part of general language), that it seems impossible to use them to auto-

matically identify specialized text to be used as reference material for a terminological task.[5]

Students were also encouraged to think about other possible problems concerning corpora which are automatically assembled from the Web. The quality and reliability of the texts (and of the terms employed in them) cannot be taken for granted; questions of register and style should be taken into account, as well as their relevance to the task.

However, the quality of a corpus ultimately depends on the quality of information the translator/terminologist is able to extract from it (Varantola 2003). Besides being used for term extraction, DIY specialized corpora can be rich sources of other information to be recorded in a terminological sheet, such as definitions, contexts, semantic relations etc.

From this point of view, all the corpora collected by the different groups turned out to be relevant to the task. Students were encouraged to look for definitions, contexts, synonyms and variants of terms with the aid of a concordancer.[6] They were, for instance, advised to search for defining expressions and linguistic signals such as "*is a kind of*", "*consists in*", "*known as*", "*also called*" etc. Some explicit definitions were present in each corpus, but it was interesting to notice that – where the need arose to infer definitions from the text – the less formal texts often proved to be more useful than the more specialized ones, possibly because of the need to explicate concepts for the less expert audience involved.

Discussion was therefore triggered about the pros and cons of

[5]Because of the time constraints of doing such activity in the classroom, we did not reiterate the bootstrapping procedure using unigrams and multi-word terms extracted from the first downloaded corpus, as suggested by Baroni and Bernardini (2004). This might have helped to retrieve more specialized texts, but it might equally have degraded the output.

[6]In this case, the IMS Corpus WorkBench (Christ 1994) was used to encode and index the students' corpora, and the associated Corpus Query Processor (CQP) was used for concordancing.

automatically building corpora from the Web: despite the drawbacks pointed out above (mainly, the lack of control over text sources, but also the incompleteness of the material – i.e., it was not possible to find definitions and useful contexts for all terms), most students stated that they were favorably impressed by the possibility of collecting such large amounts of reference materials they could use for any translation task, with such little effort and in such a short time. The group working on cell phones also realized that manually creating a corpus from the Web for their domain (i.e., "hunting" via Web queries through search engines; cf. Fletcher 2004) would be equally difficult and more time-consuming, as there is too much information online whose relevance needs to be evaluated before finding the "right" texts for a corpus like this one.

3.2 Applying experience to the end-of-course project

The final test for the course consisted in a composite project, based on the English-to-Italian translation of a text on the domain of asparagus cultivation; the source text was chosen by the teachers mainly on the basis of its degree of specialization, i.e., rich of domain-specific terms but not too technical. Students were given the source text to be translated, and were asked to collect reference corpora in both source and target language as well as to produce a given number of terminological sheets with information extracted from the corpora. Following our classroom discussions, they were let free to decide whether to build the corpus automatically or manually, and they were asked to provide feedback on the reasons underlying their choice.

As expected, all the students who have taken the exam at the time of writing decided to try and work on automatically assembled corpora. We will first analyze the procedure followed to create corpora for the source language, then moving on to target language corpora and corpus use.

Concerning the choice of the seeds on which to base the boot-

strapping process for the source language corpus, most of the students identified specialized terms within the source text and added a few more general terms, such as "*cultivation*", which were not present in the text but which were perceived to be relevant. Most of them also demonstrated an understanding of the Web mining procedure by increasing – compared to what was done in class – the number of tuples to be searched as well as the number of webpages to be retrieved for each tuple, in order to retrieve larger corpora.

As far as the target language corpus was concerned, some students reported that they had chosen the seeds by guessing – and verifying with dictionaries – potential equivalents of source terms. Two students, on the other hand, decided to use a search engine to identify some relevant and (presumably) authoritative webpages in the target language and to extract candidate seeds for the bootstrapping procedure from such pages. In one case, this proved to be a good intuition, which allowed the student to reduce the risk of "circularity" (Varantola 2003), i.e., the risk of choosing wrong (translations of) keywords and to build corpora on such unsuitable terms. In the other case, however, the suitability of the extracted terms was not evaluated carefully, and the student (a non-native speaker of Italian) ended up choosing an extremely rare word, i.e., *brattea* ("bract"), which probably spoilt the results of some automatic searches. It is important to always keep in mind the need for careful evaluation of seeds and the limitations of automatic corpus creation from the Web.

After examining their target language corpora in view of the compilation of the termbase and of the translation, however, most of the students found that their material was not sufficient to retrieve all the information needed, i.e., suitable definitions and domain-relevant contexts, and some of them decided to build another corpus semi-automatically, with the aid of a program (Text-

Stat)[7] which allows users to assemble corpora by specifying the URLs of the webpages to be downloaded, which might be either previously known or discovered through a search engine. According to their reports, this process of focusing on and downloading predetermined reliable websites, which we might call "grazing" (Fletcher 2004), proved to be very effective: not only could they evaluate the relevance and quality of texts before including them in the corpus, they could also build corpora rich in useful information while keeping them to an easily manageable size. Moreover, as many authors have already pointed out (see, e.g., Zanettin 2002, Maia 2002), the fact of having to find and read candidate reference texts prior to the translation task proper helped students to familiarize themselves with the specialized subject, thus enhancing their understanding of the domain and, possibly, of the source text; some students actually reported that visiting several websites allowed them to find pictures and images which helped them to better understand the structure of the asparagus plant.

4 Concluding remarks

The course in Terminology and LSP was designed, among other objectives, to sensitize students to the great possibilities offered by a more conscious and profitable use of a tool – i.e., the Web – with which they are already acquainted, by showing them how easy it can be nowadays to build corpora which could be used, along with traditional online dictionaries or glossaries, as performance-enhancing tools within some specific translation or terminological task.

While preparing their end-of-course projects, students realized that the advantages of automatically assembling corpora from the Web were counterbalanced by the need to carefully assess the qual-

[7] Freely downloadable from `http://www.niederlandistik.fu-berlin.de/textstat/software-en.html`

ity of the results, but that it was simple for them to use the Web itself to adjust their corpora by adding other relevant material with more or less automated methods. Nonetheless, it is only after having acquired some competence on a specific domain that it is possible to see the need for and to carry out such "corrections".

Our conclusion is therefore that the degree of usefulness of LSP corpora automatically assembled from the Web depends first and foremost on the user's familiarity with the specialized domain in question. Studying the terminology belonging to a domain which is totally – or mostly – unknown to the user through corpora created automatically can be quite risky, as the user would not have the necessary knowledge to judge the appropriateness of the output. As far as terminology is concerned, however, such output would mainly depend on the content of webpages, and less on the quality of the Web mining tool; in this respect, it might be argued that even search engine results can be difficult to interpret for the non-expert eye, the Web being rich in unreliable, non-authoritative materials. On the other hand, when the user has – or has acquired – sufficient domain-specific knowledge to be able to critically evaluate texts/terms retrieved with no – or limited – human supervision, the possibility to collect large quantities of data in such a short time cannot but prove of great value for terminologists and translators alike.

References

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. In *Proceedings of LREC 2004*, 1313-1316.

Christ, O. (1994). A modular and flexible architecture for an integrated corpus query system. *Proceedings of COMPLEX'94: 3rd Conference on Computational Lexicography and Text Research.*

Fletcher, W. (2004). Facilitating the compilation and dissemination of ad-hoc Web corpora. In Aston, G., Bernardini, S. and Stewart, D. (eds.) *Corpora and language learners*, Amsterdam: Benjamins, 273-300.

Gamper, J. and Stock, O. (1998). Corpus-based terminology. *Terminology* 5(2), 147-159.

Gavioli, L. and Zanettin, F. (1997). Comparable corpora and translation: A pedagogic perspective. *First international conference on Corpus Use and Learning to Translate.*

Maia, B. (2002). Corpora for terminology extraction: The differing perspectives and objectives of researchers, teachers and language service providers. *Language Resources for Translation Work and Research, LREC 2002 Workshop Proceedings*, 25-28.

Pearson, J. (1998). *Terms in context*, Amsterdam: Benjamins.

Pearson, J. (2000). Teaching terminology using electronic resources. In Botley, S. , McEnery A. and Wilson, A. *Multilingual corpora in teaching and research*, Amsterdam: Rodopi, 92-105.

Sager, J. (2001). Terminology compilation: Consequences and aspects of automation. In Wright, S. and Budin, G. (eds.) *Handbook of terminology management: Volume 2, application-oriented terminology management*, Amsterdam: Benjamins, 761-771.

Varantola, K. (2003). Translators and disposable corpora. In Zanettin, F., Bernardini, S., and Stewart, D. (eds.) *Corpora in translator education*, Manchester: St. Jerome, 55-70.

Zanettin, F. (2002). DIY Corpora: The WWW and the translator. In Maia, B., Haller, J. and Ulrych, M. (eds.) *Training the language services provider for the new millennium*, Porto: Universidade do Porto, 239-248.

Specialized Corpora from the Web and Term Extraction for Simultaneous Interpreters

Claudio Fantinuoli

1 Introduction

There is no doubt that the Web is a mine of language data of unprecedented richness and ease of access (Kilgarriff and Grefenstette 2003). As more people use the Web for more tasks, it provides an increasingly representative machine-readable sample of interests and activity in the world (Henzinger and Lawrence 2004). Despite some drawbacks, the Web is an immense source of disposable corpora (Varantola 2003) that can be used for specific purposes such as translation or interpretation tasks. Many language professionals use the Web as a source of information to study the language and process the specific terminology; in some cases, they also build a corpus to be looked up with a concordancer, but this is done through manual queries and downloading. Obviously this is an extremely time-consuming task. The time investment is perceived as particularly unjustified if the final result is meant to be a single-use corpus. If the aim is that of constructing a corpus big enough to allow terminology extraction, then an automated process to bootstrap corpora from the Web is the best solution to speed up the process.

When preparing themselves for a highly specialized conference, interpreters must acquire linguistic and extra-linguistic information in order to perform a good interpretation task (Gile 1995). As

Kalina (1998) points out, the elaboration of the preparatory documentation can help interpreters to advance the workload and to improve the working conditions in the booth. This preparation is nowadays very traditional, i.e., it is done manually and it includes: collection of parallel texts, reading (acquisition of extra-linguistic information) and elaboration and memorization of glossaries containing the specific terminology (language learning). This task appears to be time consuming and not efficient enough if we take into account the time factor, i.e., the time conditions under which a professional interpreter is used to work. To facilitate this process, we propose an approach to "Corpus Driven Interpreters Preparation". The process of "knowledge acquisition/language learning" needed by interpreters in order to prepare themselves for a conference can be optimized by making it "terminology-driven", or "bottom-up": from the terminology to the conceptual structure of a particular domain. Corpora can be the source of a potentially endless "serendipity process" (Johns 1988), as one word or phrase leads to another, depending on the user's intuition and individual proficiency, interests or needs. In this approach, the interpreter will "explore" the corpus starting from a list of specialized terms. In this way s/he will learn the terms, their meaning and usage in context,granting that amount of flexibility and active interaction typical of the interpreter's preparation. A list of specialized terms, the starting point of this kind of preparation, can be obtained by automatically extracting the specific terminology from a corpus. To speed up the process, the corpora can be automatically created using tools such as BootCaT (see section 2 below) and the Web as a source of specialized texts. The interpreter will then look up the corpus using a concordancer.

In this experiment we compare two procedure of terminological extraction using two different specialized corpora: the first is a manual corpus built by a terminologist in order to manually extract the specialized terms of the domain (childhood acute lym-

phoblastic leukemia), the second is a corpus automatically generated by the BootCaT tool using the Web as a corpus and a series of starting seeds that are expected to be representative of the domain under investigation. This list of seeds closely resembles what many interpreters have at their disposal as preparatory documentation in real life, i.e., the keywords of the abstracts given to the interpreters from the conference organizers. Our first aim is to evaluate BootCaT and in general the use of the Web as a corpus for specialized purposes. In our study we consider professional interpreters to be the target users of this tool. Interpreters represent a special user typology and the terminology needed varies according to the needs of the interpreter. Thus we will propose three different criteria for evaluating the tool. The experiment is conducted with Italian, German and English corpora.

2 The BootCaT procedure

In the last few years several experiments have used the BootCaT toolkit to bootstrap corpora from the Web in order to extract linguistic information such as terms or collocations. See, for example, Baroni and Bernardini (2004), Baroni and Ueyama (2004) and Sharoff (this volume). The multi-word term extraction method we implement has some similarities with the one proposed by Baroni and Bernardini (2004).

The basic BootCaT procedure is very simple.[1] Basically two main tasks are accomplished by the tool: 1) building a corpus of specialized texts from the Web; 2) extracting the relevant terminology from the downloaded corpus.

BootCaT compares frequencies in specialized and reference corpora to look for terms typical of the former. This is a fairly common idea in terminology extraction and corpus comparison

[1] For a more detailed description of the procedure see Baroni and Bernardini (2004).

work. See, for example, Rayson and Garside (2000) and Kilgarriff (2001). The tool uses an iterative algorithm to bootstrap corpora from the Web and extract unigram terms. It then proceeds to extract multi-word terms on the basis of the downloaded corpus and of the unigram term list extracted in the previous phase. The bootstrapping process, using the Google search engine,[2] starts with a small list of seeds that are expected to be representative of the domain. The seed terms are randomly combined and each combination is used as a Google query string. The top n pages (HTML, PDF and doc files) returned for each query are retrieved and formatted as text. The unigram terms are extracted from the corpus of retrieved pages by comparing the frequency of occurrence of each word in this set with its frequency of occurrence in a reference corpus. Frequencies are compared using the Mutual Information (Church and Hanks 1990) and the Log Likelihood (Dunning 1994) association measures.

To make it to the final candidate lists of simple and multi word terms, the extracted terms must fulfill two criteria: 1) they must correspond to a specific morphosyntactic pattern (section 7); 2) they must contain at least one of the extracted unigrams.

3 Empirical assessment

Evaluating the performance and the differences between the terminological extractions from an automatic downloaded corpus and a manual corpus is not an easy task. In this case, the situation is further complicated because we try to take into account a well defined potential user of the extracted data, the professional interpreter. With this in mind, we base our evaluation on: the quality of terms based on human assessment – i.e., well- or ill-formed – and on their degree of specialization; the level of specialization of

[2] `http://www.google.com/apis`

	Words	Bytes
Italian	108,016	763,455
German	88,895	738,695
English	286,346	2,037,176

Table 1. Manually collected specialized corpora

the extracted terms in light of the needs of interpreters; the comparison of the extracted terms with a reference term list manually created by a professional terminologist.

The reference term lists (RTL) were created from manually constructed corpora (see table 3) collected by a terminologist in a multilingual project on "childhood acute lymphoblastic leukemia" (Bordoni 2001). The Italian RTL contains 136 terms; the German one 158 terms; the English one 155. The collection of the texts, mainly from the Internet (PDF, doc and HTML), but also from printed papers, and the extraction procedure were all done manually, i.e., searching for suitable websites, evaluating the quality of the texts and then extracting from them the specialized terminology.

In order to make the comparison of the manual and the automatic terminology extraction methods more fair, we excluded from the manual lists the terms that were extracted from printed texts by the terminologist and were not found in her corpus. Notice, however, that we base the evaluation on terms that were extracted by the terminologist from her manually compiled corpus. Thus, when we compare the quality of term extraction between the manual and automatically constructed corpus below, we are actually giving an advantage to the manual corpus, given that we use a list of terms that were extracted from it as our golden standard.

4 Evaluation of the candidate terms

4.1 Five-level taxonomy

The candidate terms were divided into five groups according to their level of specialization and well-formedness:

1. specialized terms contained in the reference term list;
2. specialized terms not contained in the reference term list;
3. general medical terms;
4. "general" terms;
5. incomplete or ill-formed terms.

Category 1 contains terms that were manually extracted by the terminologist (and therefore are contained in the RTL), e.g.: *epatosplenomegalia*, *intrathekale Chemotherapie* and *bone marrow aspiration*. In category 2 we find highly specialized terms that were not detected by the professional terminologist, e.g.: *leucemia mieloblastica acuta*, *myeloische Leukämie* and *allogenic peripheral blut*. Category 3 contains non-specialized terms that are commonly used in the field of medicine, e.g.: *apparato urinario*, *antibiotische Therapie* and *bone*. In category 4 we find general terms that are not specific to the medical field, e.g.: *fattore*, *statistische Auswertung* and *Journal*. Category 5 contains ill-formed, incomplete expressions and fragments, e.g.: *sempre alla stessa*, *Kind selten* and *recurrent childhood*.

All extracted terms were evaluated according to this grid. Of course there is always an amount of arbitrariness in this kind of evaluation, even though we aimed for consistency: make the same judgment for the same term independently of the extraction method.

4.2 The target user: The interpreter

As Kurz (1996) points out, interpreters may need both specialized and less specialized terms in order to prepare themselves for a conference. Depending on whether the interpreter is interpreting into or out of the foreign language or whether s/he is used to interpreting in that specific domain or not, we can have two main scenarios:

a the interpreter needs only the highly specialized terms regarding the subject field (in our case leukemia); or

b the interpreter needs the specialized terms plus the more general medical terms.

4.3 Second-level taxonomy

To account for the needs of interpreters (section 4.2), the 5 categories of the original taxonomy (section 4.1) were merged in what we call T2a e T2b. To evaluate the precision of the system to extract only the highly specialized terminology of the domain, we use the taxonomy T2a:

T2a = {A1,B1} where A1 = {1,2} and B1 = {3,4,5}

A1 are the acceptable highly specialized terms, i.e., the sum of the terms belonging to category 1 (extracted terms that were also manually detected) and to category 2 (highly specialized terms that were not manually detected).

We evaluate the quality of the system in extracting terms from the medical domain – highly specialized terms and otherwise – with the taxonomy T2b:

T2b = {A2,B2} where A2 = {1,2,3} and B2 = {4,5}

A2 are the acceptable medical terms, i.e., the sum of the terms belonging to group 1 (extracted terms that were also manually detected), to group 2 (terms specific to the domain that were not manually detected) and to group 3 (generic medical terms).

4.4 Recall

We evaluate terminological extraction from the two different corpora in terms of precision and recall, using the two taxonomies just described. In our study we define *Recall* (for category 1) as follows:

$$Recall = \frac{AUTOTERMS}{MANTERMS} \times 100$$

$AUTOTERMS$ is the number of category 1 terms that were automatically extracted, and $MANTERMS$ the number of terms manually identified by the terminologist.

We consider the manually extracted terms as being the only terms contained in the corpora and compute the recall value on the number of terms retrieved manually. The recall gives us an idea of the amount of terms contained in the reference terminology list (the one compiled by the terminologist) that are retrieved by the semi-automatic system. In this way we compare the results of the manual and the automatic term extraction procedures (given that recall is based on terms that were extracted from the manual corpus, we would expect, in principle, higher recall when automated extraction is performed on the manual corpus).

5 Corpus construction

We started the bootstrapping process with a series of 9 seeds for each language (table 2). As far as interpreters are concerned, we can suppose that the initial terms can be obtained from the

Italian	German	English
leucemia	Leukämie	Leukemia
“midollo osseo”	Knochenmark	“bone marrow”
LLA	ALL	ALL
chemoterapia	Chemoterapie	chemotherapy
trapianto	Transplantation	transplantation
“leucemia acuta linfoblastica”	“akute lymphatische Leukämie”	“acute lymphoblastic leukemia”
linfocita	Lymphozyt	Lymphocyte
“puntura lombare”	Liquorpunktion	“lumbar puncture”
leucociti	Leukozyten	Leukocytes

Table 2. Initial seeds used to create the corpora

	Italian	German	English
URLs	308	128	304
Bytes	12,519,130	7,555,510	3,086,908

Table 3. Number of URLs and size of the corpora

conference abstracts delivered to the interpreter. Note that in order to grant similar initial conditions for all languages, we used the translation of the same seeds in every extraction.

As BootCaT allows the user to control several important parameters, such as the number of queries issued for each iteration, the number of seeds used in a single query, the number of pages to be retrieved, etc., we downloaded files using the following parameters: 3 seeds for each query; 20 tuples, each used for a query; a maximum of 20 pages to be downloaded for each query. The number of URLs, without counting duplicates, obtained with this method is shown in table 3. Then we proceeded by automatically downloading and converting the detected URLs into text files; the size of the corresponding corpora is also reported in table 3.

The size of the downloaded corpora varies considerably among the languages and this even though the initial conditions were

	Italian	German	English
Reference corpus	3,288,496	3,109,525	3,388,390
Specialized corpus (Web)	1,512,766	813,817	422,037
Specialized corpus (manual)	105,890	85,074	274,215

Table 4. Size of the corpora in tokens

virtually the same for all extractions (seeds and BootCaT parameters). Interesting enough, the language with the least amount of text is English, the language of international scientific communication.

6 Extraction of unigram terms

We first tokenized the specialized and the reference corpora with command-line scripts (table 4).

The reference corpora are part of the EuroParl corpus, a large collection of texts from the European Union.[3] They cover a large variety of topics and this makes them suitable to be used as a benchmark for corpus comparison. Using the UCS tools,[4] we compared frequencies between the reference and the specialized corpora. We computed both the Mutual Information (MI) and the Log-Likelihood (LL) association measures in order to account for terms with low and high frequencies (Evert and Krenn 2001). In our experiment MI and LL are not used to compute the proximity factor of two words in a given text (the probability that a word occurs with another word – collocation), but to compare the occurrences of a given word in two different corpora, as illustrated by Sharoff (this volume). We extracted the final unigram term lists considering only the first 200 words obtained with every

[3] `http://people.csail.mit.edu/people/koehn/publications/europarl/`

[4] `http://www.collocations.de`

	Italian	German	English
corpus (Web)	390	355	298
corpus (manual)	409	399	468

Table 5. Size of unigram lists

Italian	German	English
N+ADJ+ADJ	ADJ+ADJ+N	ADJ+ADJ+N
N+ADJ	ADJ+N	ADJ+N
N	N	N
N+N		N+N
N+PRE+N		N+N+N

Table 6. Morphosyntactic patterns

association measure. In addition we extracted acronyms simply by searching for capital letter words longer than 1 and shorter than 4 characters. We merged the three lists obtaining the numbers of candidate unigram terms reported in table 5 (some examples: for Italian, *anemia, induzione, EFS, leucociti, citogenetica*; for German, *B-ALL, Blasten, Blutbild, Chemoterapie, Erbrechen*; for English, *cyclophosphamide, cyclosporine, cytarabine, leukemia, MRD*).

7 Extraction of multi-word terms

The unigram lists and the corpora were used to extract multi-word terms. We first tagged the specialized corpora using the TreeTagger[5] and then built bigrams and trigrams.

We extracted multi-words terms that satisfied the POS patterns shown in table 6 and that contained at least one unigram from the lists previously extracted (section 6).

[5] `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTree/Tagger.html`

	Italian	German	English
corpus (Web)	353	333	317
corpus (manual)	290	324	335

Table 7. Candidate terms

Taxonomy	Extracted terms	%	Recall
1	13	3.68	9.56%
2	85	24.08	
3	201	56.94	
4	30	8.5	
5	24	6.8	
Tot terms	353	100	

Table 8. Results (Web): Italian

The lists of single and multi word terms were then merged (table 7).

8 Evaluation

8.1 General

We assigned a value to every candidate term according to our taxonomy (section 4.1). As pointed out above, we focused primarily on consistency. We manually assigned each term to a category of our grid (the terms were evaluated in random order and without knowing their source). For the five categories we obtained the results reported in tables from 8 to 13 (while reading these tables, please keep in mind our categorization from section 4.1 – 1: Specialized terms contained in the reference term list; 2: Specialized terms not contained in the reference term list; 3: General medical terms; 4: General terms; 5: incomplete or ill-formed terms).

As far as the 5 categories are concerned, we can easily see that there are similarities among the languages. The obvious difference

Taxonomy	Extracted terms	%	Recall
1	50	15.01	32.64%
2	145	43.54	
3	78	23.42	
4	35	10.51	
5	25	7.5	
Tot terms	333	99.98	

Table 9. Results (Web): German

Taxonomy	Extracted terms	%	Recall
1	48	15.14	30.97%
2	139	43.85	
3	87	27.44	
4	30	9.46	
5	13	4.1	
Tot terms	317	99.99	

Table 10. Results (Web): English

Taxonomy	Extracted terms	%	Recall
1	59	20.34	43.38%
2	91	31.38	
3	77	26.55	
4	57	19.65	
5	6	2.07	
Tot terms	290	99.99	

Table 11. Results (manual corpus): Italian

Taxonomy	Extracted terms	%	Recall
1	53	16.36	33.54%
2	139	42.9	
3	33	10.18	
4	57	19.65	
5	21	6.48	
Tot terms	324	99.99	

Table 12. Results (manual corpus): German

Taxonomy	Extracted terms	%	Recall
1	38	11.34	24.51%
2	152	45.37	
3	91	27.16	
4	38	11.34	
5	16	4.78	
Tot terms	335	99.99	

Table 13. Results (manual corpus): English

concerns the value obtained for the Italian automatically downloaded corpus. If we pay attention to the distribution of terms within Italian, we see that most terms are in the third category, i.e., general medical terms. This means that the downloaded Italian corpus is less specialized then the German and the English ones, even though the initial seeds were the same. Again, this is an interesting starting point to further investigate differences in Web document availability in different languages.

If we consider the recall values, we see that the automatic extraction of highly specialized terms from the downloaded corpora leaves out many terms that were considered important by the terminologist. While this may cast some shadows upon the effectiveness of the automatic method of terminology extraction used (from the terminologist's prospective), it does highlight the fact that both corpora – manual and automatic – are of comparable quality (from the extraction's perspective). This is especially interesting since the manual set used for recall assessment was extracted from the manual corpora, thus we know that the manual set terms are present in the latter, that, in principle, should thus provide higher recall than the automatically constructed corpora.

	Extraction from Web corpus	Extraction from manual corpus
	Italian	
A1	27.76	51.03
A2	84.70	78.27
	German	
A1	58.55	59.26
A2	81.97	83.33
	English	
A1	58.99	56.71
A2	86.43	83.87

Table 14. Comparison between A1 and A2 (in percentage)

8.2 Interpreter-targeted evaluation

As we pointed out before, the ultimate criteria to evaluate the tool are the needs of professional interpreters. This is why we evaluate it according to the taxonomies T2a and T2b, i.e., according to the capacity to extract highly specialized terms (A1) or specialized terms plus the more general medical terms (A2).

The results (table 14) are similar across the different languages, besides the expected exception of A1 with the Web corpus in Italian. For the category A1 – specialized medical terms – the best result was obtained with the manual corpus for the German language (59.26%). But the results obtained with the Web corpus are very close to this value: German 58.55% and English 58.99%, the latter being the best result obtained with this language. For the category A2 – specialized and generic medical terms – the best result was obtained with a Web-derived corpus (English, 86.43%).

Again, these results have to be interpreted by keeping in mind that a portion of the terms in A1 and A2 (namely the terms in the manual set) have been extracted from the manual corpus, which is, thus, advantaged in terms of the evaluation procedure.

9 Conclusion

We showed that term extraction from manually compiled and automated Web-derived corpora leads, in general, to comparable results (further research is needed on the reasons for poor performance of the Web-based procedure in Italian).

Given how time-consuming it is to build a corpus by hand, automated Web-based corpus construction is a very promising way to reach good results with limited efforts.

Using the BootCaT procedure, interpreters preparing for a conference can obtain a list of relevant terms and texts within minutes, even when targeted preparatory materials have not been made available by the conference organizers (as is often the case in professional settings). While the current version of the BootCaT toolkit requires computational skills beyond what is reasonable to expect from interpreters, the graphical interface currently being tested (Baroni et al. 2006) has the potential to make BootCaT a very popular tool for our target community.

References

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. *Proceedings of LREC 2004*, 1313-1316.

Baroni, M., Kilgarriff, A., Pomikálek, J., Rychlý, P. (2006). WebBootCaT: Instant domain-specific corpora to support human translators. *Proceedings of EAMT 2006*, 247-252.

Baroni, M. and Ueyama, M. (2004). Retrieving Japanese specialized terms and corpora from the World Wide Web. *Proceedings of KONVENS 2004*.

Bordoni, F. (2001). *Leucemia linfoblastica acuta in età pediatrica:*

Proposta di glossario terminologico trilingue (italiano - tedesco - inglese). Unpublished dissertation, SSLMIT, Bologna.

Church, K. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics* 16(1), 22-29.

Dunning, T. (1994). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics* 19(1), 61-74.

Evert, S. and Krenn, B. (2001). Methods for the quantitative evaluation of lexical association measures. *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, 188-195.

Gile, D. (1995). *Basic concepts and models for translator and interpreter training*, Amsterdam: Benjamins.

Henzinger, M. and Lawrence, S. (2004). Extracting knowledge from the World Wide Web. *Proceedings of the National Academy of Sciences of the United States of America* 101, 5186-5191.

Kalina, S. (1998). Kognitive Verarbeitungsprozesse. In Snell-Hornby, M., Hönig, H., KuSSmaul, P. and Schmitt, P. (eds.) *Handbuch Translation*, Tübingen: Stauffenburg, 330-335.

Kilgarriff, A. (2001). Comparing corpora. *International Journal of Corpus Linguistics* 6, 1-37.

Kilgarriff, A. and Grefenstette, G. (2003). Introduction to the special issue on the Web as corpus. *Computational Linguistics* 29(3), 333-347.

Kurz, I. (1996). *Simultandolmetschen als Gegenstand der interdisziplinären Forschung*, Wien: WUV-Univ. Verlag.

Johns, T. (1988). Whence and whither classroom concordancing? In Bongaerts, T., de Haan, P., Lobbe, S., and Wekker H. (eds.) *Computer applications in language learning*, Dordrecht: Foris, 9-27.

Rayson, P. and Garside, R. (2000). Comparing corpora using frequency profiling. *Proceedings of Workshop on Comparing Corpora at ACL 2000*, 1-6.

Varantola, K. (2003). Translators and disposable corpora. In Zanettin, F., Bernardini, S. and Stewart, D. (eds.) *Corpora in translator education*, Manchester: St. Jerome, 55-70.

The Net for the Graphs: Towards Webgenre Representation for Corpus Linguistic Studies

Alexander Mehler *Rüdiger Gleim*

1 Introduction

In recent years, the Web has become increasingly significant for corpus linguistic research (Baroni and Bernardini 2004; Keller and Lapata 2003; Kilgarriff and Grefenstette 2003; Resnik and Smith 2003; Santamaría et al. 2003). On the one hand, it contains a vast amount of hypertext documents of newly emerging document types (e.g., *conference websites*, *corporate sites*, *electronic encyclopedias*, *hotlists*, *sites of online shops*, *(personal, academic) home pages*, *weblogs* etc.). On the other hand, the Web has become accepted as a common platform for information exchange so that one can find instances of almost any type of electronic text imaginable. This, *in theory*, makes the Web the source of choice when large corpora for studying language varieties are needed. But it also makes it the source of choice when studying the emergence and evolvement of hypertext types. The reason for this assessment is also the main source of difficulties one has to face following this line of research: Web-based hypertext authoring mostly utilizes languages, as for example HTML, CSS and related "standards", in spite of their well-known deficits regarding the separation of structure, content and form. Moreover, these languages do not at all standardize the content-based, functional structuring of websites, neither with respect to the internal structuring of constitu-

tive webpages, nor with respect to page linkage. Rather, the kinds of structuring and linkage observable on the Web emerged spontaneously and rapidly during its short history. These kinds are, of course, not completely determined by the medium or authoring software used, but vary with the different functions and contents they carry and the styles of Web document authors. Nevertheless, hypertextual patterns allow reliable predictions of the functions being manifested. We have no problem distinguishing, for example, a personal academic home page from a conference website not only in terms of content but also in terms of *document structure*.

The Web apparently manifests an evolution of hypertextual patterns *in fast motion* making its various mutations accessible to corpus linguistic studies. This implies that the tremendous differences in structural quality manifested by websites are by no means a venial deficit to be abstracted away by hypertext representation. Rather, this informational variety is an indispensable characteristic of the kind of structure formation under consideration. As a consequence, any approach to representing Web-based patterns of hypertext authoring has to face the task of representing and processing various aspects of informational uncertainty. In other words: The apparatus of probabilistic modeling will be needed in order to model, for example, aspects of structural ambiguity, under-specification and vagueness of structural descriptions of Web-based units, their constituency and dependency structure.

This paper is about prerequisites of representing patterns of Web-based hypertext authoring. Its basic tenet is that *websites* and their constitutive *pages* are instances of *webgenres* (Crowston and Kwasnik 2003; Crowston and Williams 1999, 2000; Dillon and Gushrowski 2000; Orlikowski and Yates 1994; Rehm 2002; Yoshioka and Herman 2000) and their elementary *stages* (Ventola 1987) or *phases* (Eggins 1994) by analogy with texts and their components as instances of genres and generic stages (Martin 1992). We hypothesize a webgenre to be identifiable by means of function

bearing patterns whose variance within the same genre is lower than between different ones. In the following sections, we discuss an indispensable prerequisite for automatically studying this functional variety, namely the download and representation of presumptive webgenre instances on the level of websites.

When it comes to an experiment in corpus-based analysis in this area, one is confronted with a tremendous set of problems. To name only a few of these: How do we identify the extent of a website of a given webgenre? In other words, how do we identify Web-based hypertext borders? What does an appropriate representation model look like which allows one to represent the different kinds of textual and hypertextual structures manifested by websites? How do we deal with flawed website manifestations as a result of, for example, malformed HTML-coding, broken links or missing structural explicitness? How do we make the resulting website representation retrievable for the different tasks in corpus linguistic research?

Since a loss of information occurs every time a website is taken out of its context, answers to these questions have to be carefully considered. We hypothesize that appropriate answers get their validity to the degree to which they clarify the relation of *explicit* (*visible*) or *manifesting* website structure and *implicit* (*hidden*) or *manifested* webgenre structure.

This paper addresses some aspects of representing hypertextual units with a focus on websites as instances of webgenres. The subsequent sections concentrate on representational and technological issues of this task. Starting from a draft of our *conceptual data model* of webgenres, some major problems in website representation are specified in section 2. This relates, amongst other things, to the so called *polymorphism* and *polyfunctionality* of hypertextual units. In section 3 we sketch our *logical data model* which is based mainly on graph theory. Subsequent to this logical specification of the conceptual model, its physical implementation

is presented in section 3 too. We utilize the G*raph* e*X*c*hange* L*anguage* (GXL) (Winter et al. 2002) and thus propose a document schema as an appropriate format for physical data modeling of websites. As this paper focuses on the *explicit* (*visible*) structure of websites, sections concentrate on representing hyperlinks (3.1), the nesting of link, document and linguistic structure (3.2) and structure formation in time (3.3). Section 4 utilizes this model in order to derive constraints for exploratory corpus analyses. Finally, the conclusion gives a prospect on future work. This relates especially to *mining* and representing the implicit genre-specific, functional structure of websites. In summary, the present paper can be seen as a preparatory step towards mining this hidden webgenre structure.

2 Outline of a conceptual model of genre-specific website structuring

According to discourse analysis, distributional patterns vary depending on the functions of the discourses in which they are observed (Biber 1995). Starting from the *weak contextual hypothesis* of Miller and Charles (1991) which says that the similarity of the contextual representations of words contributes to their semantic similarity, one might state that differences of textual form reflect differences in function as far as they are confirmed by a significantly high number of instances and thus are recognizable as text patterns. The main objective of the approach followed by the present paper is to verify this hypothesis in the area of Web-based documents. That is, we expect websites of different genres to be distinguished by the function bearing patterns they manifest. We expect this distinguishability to also hold – although to a minor degree – for the constituents of websites (e.g., webpages) and the sub-functions they serve.

In order to further specify this hypothesis, the concept of *web-*

genre has to be narrowed down. This can be done by abstractly defining a *document class* as a class of textual or hypertextual units which serve the same or related functions and thus manifest similar structures and layout shapes. Different criteria of document class formation relate to different types of access to such functional entireties. If we consider, for example, the composition of classes from an extensional point of view, that is from the point of view of their document elements, we deal with *text sorts* (Heinemann 2000). If we concentrate instead on *situative* or *communicative* criteria of class membership, we deal with *registers* (Biber 1995; Halliday and Hasan 1989) and *genres* (Martin 1992), respectively. In analogy to this, we find references to *hypertext sorts*, *digital genres* and *webgenres* in case of classes of hypertextual documents (Dillon and Gushrowski 2000; Jakobs 2003; Orlikowski and Yates 1994; Rehm 2002). If in contrast to this, class membership is defined in intensional terms, we deal with *text patterns* and *superstructures* as prototypical representations of class members, whose expectation driven production/reception they support (Heinemann 2000; van Dijk and Kintsch 1983). The basis of all these approaches is the notion that structure and shape of (hyper-)textual units vary (though not deterministically) in dependence on the communicative situation or function they manifest. If we focus on structure abstracting from shape or layout, respectively, we deal with the *logical document structure*. As we deal with hypertextual units we speak, more specifically, of the logical *hypertext* document structure.

The taxonomic notion of genre of Yates and Orlikowski (1992), to which the majority of approaches to webgenres refers, aims at genre classifications. A review of the notion of webgenre is given by Firth and Lawrence (2003). They analogously identify the focus of research in this area with classification. Crowston and Williams (2000), for example, identify *hotlists*, *home pages* and *Web server statistics* as original webgenres without precursors in

literary language (Dillon and Gushrowski 2000), whose classification necessarily includes hypertextual genre markers (Crowston and Williams 1999). Consequently, the identification of sufficiently selective markers is seen as one of the main tasks of webgenre analysis (Rehm 2002). An instance of taxonomic genre analysis on the level of webpages is given by Yoshioka and Herman (2000) who analyze a single website by mapping its constitutive pages on a set of genre categories. See also Rehm (2002) who classifies generic modules of single pages.

In addition to the taxonomic notion, the procedural organization of genres is examined in systemic-functional linguistics (Halliday and Hasan 1989; Martin 1992). That is, dependency relations of generic constituents (i.e., stages or phases) and their chronology are studied from the point of view of text type formation (Ventola 1987). This approach is adopted in the present paper since it allows to identify links between pages of the same site as manifestations of webgenre internal structure (Mehler et al. 2004; Mehler and Gleim 2005). This notion is confronted with serious problems of hypertext representation which can all be traced back to the fundamental distinction of visible or manifesting website structure and hidden or manifested webgenre structure. In order to explain this, we start from a four level model of Web-based structure formation, that is of logical hypertext document structure, including the level of elementary building blocks, module types, Web document types and document network types (Mehler and Gleim 2005). *Building blocks* (manifested, for example, by tables or paragraphs) exist only as dependent parts of *module types* which relate to functionally homogeneous sub-functions of Web-based communication (e.g., *call for papers*, *program* or *conference venue* as sub-functions of the spanning function of *Web-based conference organization*).[1] Next, *Web document types* classify Web-

[1] See Storrer (2002) for a definition of the notion of module in the context of hypertext authoring.

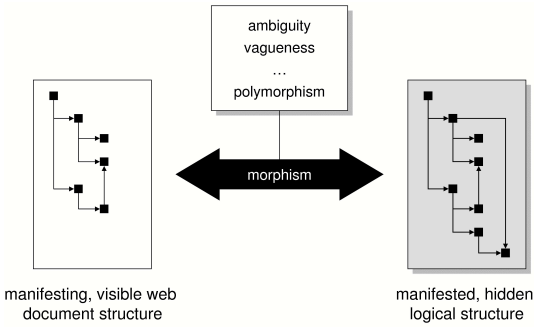

Figure 1. Informational uncertainty of the morphism interrelating manifested and manifesting structure

based manifestations of pragmatically closed acts of Web-based communication, where each of these acts serves a complex function of, for example, *conference organization*, *personal presentation* or *online shopping*. Fourth, *document network types* relate to systems of pragmatically closed, though not necessarily homogeneous communication acts. A document network type is manifested, for example, by a university's website which covers, amongst others, personal academic home pages, project sites and library sites which together contribute to the same corporate identity.

This enumeration might suggest that the levels are deterministically separated without recourse to informational uncertainty. It might also suggest that they directly relate to HTML-elements, webpages, websites and compound websites, respectively. This is, of course, not the case. In fact, there exists a many-to-many relation between functionally specified levels of Web-based communication and their manifestations by means of pages and related expression units (see figure 1), that is, between hidden hypertext document structure and manifesting website structure. Without systematizing the morphism of figure 1 – for more details see Mehler and Gleim (2005); Mehler et al. (2005) –, we only emphasize two aspects of informational uncertainty:

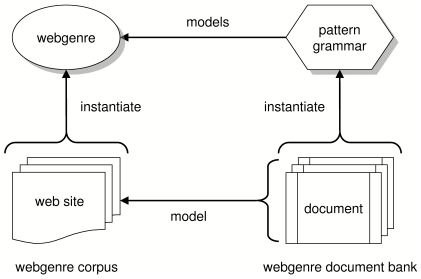

Figure 2. The basic model of document pattern-oriented webgenre analysis

- *Polymorphism* occurs if the same expression unit manifests several categories by means of separate segments. Polymorphism is given when, for example, the same webpage of a conference website provides information about the *call for papers*, the *submission procedure* and *conference registration*, that is, when it manifests two or more functions. Polymorphism results in *multiple categorizations* without being reducible to ambiguity of category assignment since in this case several categories are actually manifested by the same expression unit. Thus, resolving polymorphism cannot be reduced to the task of disambiguating category assignment as applied in machine learning and related areas.

- *Discontinuous manifestation* occurs if the same function or content unit is distributed over several expression units. Discontinuous manifestation results in *flawed* or even *missing categorizations* since in this case the webpages under consideration manifest the focal content/function category only in part. Thus, discontinuous manifestation relates to vagueness.

These two relations constitute a many-to-many relation of *function* (and *content*) units on the one hand and *expression* units on the other hand. As a result of this relation, the function or

content structure of a website is generally *not* directly accessible by just segmenting and subsequently categorizing its constitutive webpages in separation (for more information on this argumentation see Mehler et al. 2005). Moreover, links cannot be directly identified as manifestations of the "staging" of a webgenre or of the ordered progression of its phases and their structuring. This observation makes the representation of a webpage's internal *and* external structure an indispensable prerequisite for any effort in exploring the genre-specific structure of websites.

Figure 2 summarizes our webgenre model presented so far: Webgenres are considered to be manifested by websites (consisting of at least one webpage) whose structure is an informationally uncertain map of the underlying, hidden functional (webgenre) structure. As a consequence, corpora of website representations, henceforth called *webgenre document banks*, are needed, whose document elements map both: the manifesting website structure and the manifested webgenre structure as it is instantiated by the former. Finally, webgenre pattern grammars have to be induced on the basis of the input document banks which allow to classify newly observed instances according to the genre-specific patterns they manifest.

In the next section we present our approach as far as it focuses on the prerequisite of representing websites as expression units. Thus, it concentrates on the representation of explicit, visible website structure leaving the induction of the hidden logical hypertext document structures to future work (cf. Mehler et al. 2005 for a first approach to such an induction algorithm).

3 A text technological view on representing websites

This section outlines the basic building blocks of the format we use for representing websites. It is part of the `HyGraph` system

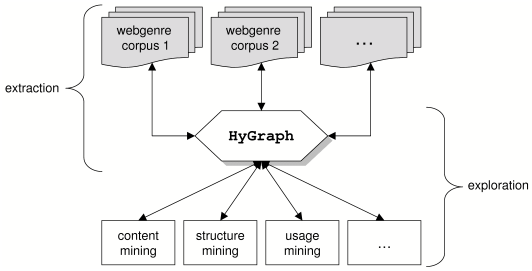

Figure 3. The HyGraph system as a generic Web mining interface for webgenre analysis

(Gleim 2005) which mediates between webgenre corpora and their processing for the various tasks of Web content, structure and usage mining (see figure 3). The HyGraph system addresses the following tasks of hypertext document processing: extraction of corpora of websites of certain webgenres; generic representation of Web documents; Web corpus management and maintenance; visualization of Web document structure; unsupervised learning of hypertext graphs.

In this paper we concentrate on the second of these tasks and thus ask for an appropriate representation format. A common framework for representing hypertextual units is graph theory. This relates especially to the area of *directed graphs*.[2] Consequently, various metrics of hypertext structure have been defined on digraphs (Botafogo et al. 1992; Chakrabarti 2002; Furner et al. 1996). However, even simple Web-based units show a structural complexity beyond digraphs. Hyperlinks, for example, often address sections of their corresponding target pages. In such relations, up to four elements can be involved: The source and target page as well as the source and target anchor. It is evident that

[2]A directed graph or digraph G is an ordered pair $G = (V, E)$ of a set V of vertices and a set E of edges where $E \subseteq V^2$. For a detailed introduction to graph theory see Melnikov et al. (1994).

this is only a simple example of many more complex cases where the expressive power of digraphs is exceeded:

- *Link structure:* Website internal and external links have to be identified as well as the graph structures (e.g., sequences, hierarchies and networks of interlinked units) they induce. In section 3.1 we consider different types of hyperlinks and the hierarchical structures they induce and transcend, respectively.

- *Nested structures:* Link classification is a new task in machine learning (Getoor 2003). It asks for representation models which go down to the wording of single pages – comparable to the bag-of-words model, but with the important difference that now graphs of such representations are needed since webpages are embedded as vertices into hypertext graphs. In section 3.2, we consider the HTML-based DOM structure and the text-based logical document structure of single pages as complements of their internal link structure.

- *Time alignment:* Websites are, of course, no stable units, but evolve in time. When they are created, conference websites often only consist of a single page announcing the conference. Then, they gradually grow as the conference approaches. Once it is over, some of the website's sections are removed (e.g., registration), others are added (e.g., conference pictures) before the website is finally deleted. In order to grasp this kind of life cycle-based structure formation, a format is needed which allows identifying different graph representations as being manifestations of the same logical unit *at different points in time*. This is outlined in section 3.3.

It is evident that a rather complex class of graphs is needed

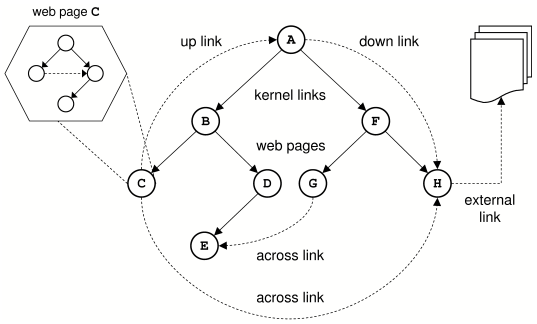

Figure 4. Types of links connecting webpages symbolized as circles

as a *logical data model* in order to meet these requirements for adequate hypertext representation. It should allow to express relations between arbitrary numbers of vertices as well as hierarchical embeddings of graphs into vertices. We utilize the G*raph* e*X**change* L*anguage* (GXL; Winter et al. 2002) as a format of *physical data modeling* in order to serve these needs. We propose using GXL for computer-based storage, maintenance and retrieval of genre-specific website representations. On the level of logical data modeling it corresponds to certain classes of graphs whose usage will also be motivated.

3.1 Representing internal and external link structure

In order to introduce our format of website representation, we start from a simplified model consisting of a directed tree (henceforth called *kernel hierarchy*) rooted by the so called leader in the sense of Eiron and McCurley (2003) (i.e., its "start page") and augmented by across, up and down links which together span a website's *hypertext graph* (see figure 4). In this section, we explain why this hypertext graph is a *hypergraph*, but not just a digraph.

The notion of a kernel hierarchy is exemplified by a conference

website headed by a menu and title page referring to, for example, its call for papers which in turn may be continued by a page on the conference's sessions etc. so that finally a hierarchical structure evolves. It is evident that the kernel hierarchy reflects navigational constraints. That is, the position of a page in this tree can be seen as reflecting the probability to be navigated by a reader starting from the root and following only its kernel links. The welcome page of a corporate website, for example, is far easier to reach than the contact information of the service hotline.

Variable	**Value**
number of websites	1,096
number of webpages	50,943
number of hyperlinks	303,278
maximum depth	23
maximum width	1,035
average size	46
average width	38
average height	3

Table 1. A sample corpus of 1,096 conference and workshop websites

A website's kernel hierarchy is spanned by so called kernel links. Kernel links have to be distinguished from across, up, down, inside and outside links (Amitay et al. 2003; Eiron and McCurley 2003; Routledge et al. 2000), which in the following are defined on the basis of the kernel hierarchy of the hypertext graph (see figure 4):

- *Kernel links* associate dominating nodes with their immediately dominated successor nodes in terms of the kernel hierarchy.
- *Down links* associate nodes with one of their (normally mediately) dominated successor nodes in terms of the kernel hierarchy – possibly parallel to a kernel link.
- *Up links* analogously associate nodes of the kernel hierarchy

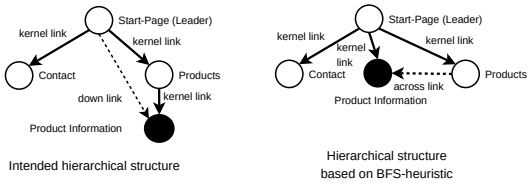

Figure 5. A problem of the heuristics of breadth first search regarding the detection of a website's kernel hierarchy

with one of their (normally mediately dominating) predecessor nodes.

- *Across links* associate nodes of the kernel hierarchy none of which is an (im-)mediate predecessor of the other in terms of the kernel hierarchy.
- *Inside links* are node (i.e., page) internal links.
- *Outside links* associate nodes of the kernel hierarchy with nodes of other websites.

Table 1 lists the frequencies of these link types as found in our test corpus of 50,943 pages of 1,096 conference websites from the fields of computer science and mathematics.

As these types of links are not explicitly tagged, they have to be automatically detected. We use a heuristic method based on a breadth-first search starting with the leader of the input hypertext graph. Consequently, pages directly accessible from the root are mapped onto the second level of the kernel hierarchy and so on until the levels of the leaves are reached. It is easy to conceive cases where this method fails to detect the correct kernel structure. If, for example, a company releases a new product there might be a newsflash on the welcome page of its website which directly links to the product description. In this case, the product description page is rated too high because of being directly accessible from the root.

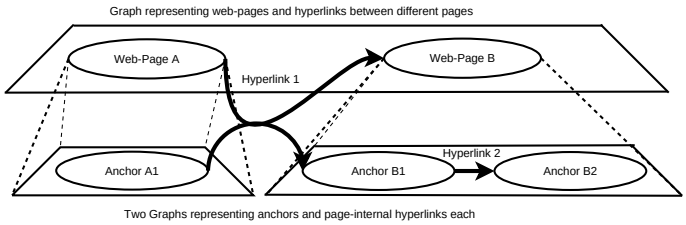

Figure 6. A layer-model of website representation embedding two page representation graphs into a website representation graph

Instead of that it should be located below the "products" page. Figure 5 illustrates this example. In order to solve this problem, it is necessary to have knowledge of the contents and purposes of webpages and of the prototypical structure of the webgenre they instantiate. That is, this example already leads to the level of implicit hypertext document structure.

The picture of website structuring we get from these considerations is that of a hypertext graph representing pages and their links as nodes and edges, respectively. As internal links belong to single pages they are represented as part of these pages' node representations (see figure 6; see also table 2). This model now allows us to introduce the physical data model based on GXL:

- *Graphs* are ordered pairs (V, E) of a vertex set V and an edge set E. In GXL, vertices are referred to as XML-elements named `node`. In the present framework, instances of this element are commonly used to represent single webpages identified by an `ID` (see table 2) and a GXL-attribute named `URI`. Accordingly, instances of the elements `edge` and `rel`(ation) are used to represent links of these nodes (see table 2).

- *Typed graphs* are graphs with typed vertices and edges. Amongst other things, we utilize typing to distinguish anchor and page nodes as well as frame source links from "standard"

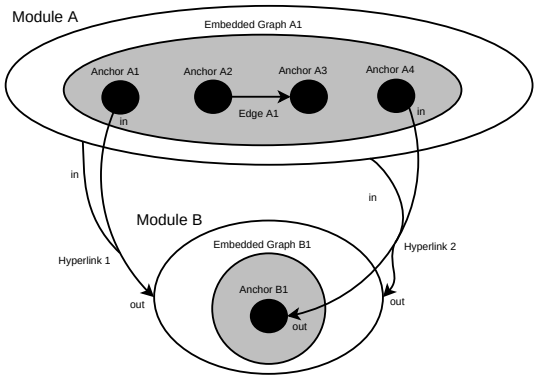

Figure 7. Three cases of page linkage (edge A1, hyperlink 1 and 2)

links. This typing (not to be confused with the distinction of link types above) is manifested by the `type` element and its `xlink:href` attribute. Since we need several type systems to independently classify the same set of hypertext constituents, we also construct attributed graphs.

- *Attributed graphs* are graphs whose nodes and edges are assigned possibly nested bags, sets, tuples or sequences of boolean, integer, real or string valued attributes. We use attributed graphs to model the URL of a webpage as an attribute-value pair and its metatags as a bag of such pairs enclosed by an instance of the GXL-attribute `MetaTags`. Unlike in Mehler et al. (2004), we do no longer map a page's textual content onto a token vector attribute, but map it as a graph on its own (see section 3.2). But we still use attributes in order to type links. That is, links are assigned a GXL-attribute `types` whose values distinguish, amongst others, between *across*, *up*, *down*, *inside* and *outside* links (see table 2).

- *Directed graphs* are graphs whose edges are ordered pairs of nodes, *adjacent from* their source node and *adjacent to* their target node. They are the default means of representing HTML links whose source and target anchors belong to the same webpage, i.e., page internal links (see link `Edge A1` in figure 7). This is done with the help of two attributes assigned to the `edge` element (see table 2): `from` and `to` take the `ID` of the corresponding source and target node anchor as values, respectively. In spite of this preferred usage, `edge` elements, their attributes and content model are not restricted to map HTML links. According to the GXL model of hypergraphs (see the last bullet of this listing), even sophisticated links following the XLink standard can be modeled by means of GXL.

- *Ordered graphs* are directed graphs whose arcs are assigned ordinal numbers reflecting any order dependent on their respective source node. In linguistics, these numbers can be used to model the syntagmatic order of the immediate constituents of the same superordinate node. In hypertext representation, they are analogously used to model the order of links which are adjacent *from* the same node. This order depends on the syntagmatic order of the links' anchors. It is manifested by means of an attribute of name `startorder` or `endorder`, respectively, which is assigned to `rel`(ation)`end` elements of the focal `rel`(ation) element.[3] All `startorder` (`endorder`) attribute values of `rel`(ation)`end`s which are incident from (to) the same node have to define a proper ordering on the `rel`(ations) involved (see table 2).[4]

[3] In the case of edges, the attributes `fromorder` and `toorder` are used instead.

[4] This is not the standard interpretation of both attributes in GXL, but the one which is needed in order to map the order of `rel`(ations) according to the syntagmatic order of the anchor nodes of the hyperlinks they are used to

- *Stratified graphs* are graphs whose nodes embed graphs on their own. In the present framework they serve to model page-internal link structures based on links whose source and target anchors belong to the same page (e.g., `Edge A1` in figure 7). In order to map the internal link structure of a page A, we embed the graph spanned by this structure into the node representing A. This part of the model is in accordance with the paradigm of document-oriented modelling complementing the predominant data-oriented character of GXL. Since page-internal links simply consist of a possibly attributed association of two anchor nodes of the same page, the `edge` element suffices as the GXL analogue of edges in digraphs in order to model this kind of link. In the case of all other links, *hyperedges* of *hypergraphs* are used instead.

- *Hypergraphs* are graphs whose *hyper*edges are subsets of the vertex set V. Hyperedges may also be ordered and directed. This qualifies them for modeling HTML links whose anchors belong to different webpages (see `Hyperlink 2` in figure 7). Table 2 illustrates an instance of the element `rel`(ation) which models a link of two pages (identified by `ModuleA` and `ModuleB`). The content model of the hyperedge in question comprises a `rel`(ation)`end` element targeting at `ModuleA` as its `sourcepage`, a `relend` targeting at `ModuleB` as its `targetpage`, and a `relend` element targeting at the link's source page anchor. Links with a target anchor specification in the URL value of their `href` attribute are modeled as `rel` elements with an additional `relend` element of role `targetanchor` (see link `Hyperlink 2` in figure 7 and table 2). Since relation ends can be extended by any GXL-attribute and since hyperedges of this kind are not restricted regard-

map. Note further that, for the time being, neither the GXL DTD nor the GXL Schema does check compliance to the latter restriction which has to be ensured by the HyGraph system.

ing the number of their targets, they allow modeling any relation of any valency. In other words, hyperedges are the preferred means of representing links, whether simple HTML links or more complex links of the XLink standard.

According to the hypertext graph model presented so far, Web-based hypertexts are represented as typed, attributed, directed, ordered hypergraphs supplemented by graph stratification and markup of the kernel hierarchy. This leaves out how to represent a page's internal content beyond its internal link structure. How this kind of graph embedding is performed is outlined in the next section.

3.2 Nesting hypertext document structures

The previous section focused on link structure representation. We have emphasized that it is necessary to distinguish layers for representing page internal and page external linkage. This leaves unspecified how to represent the remaining building blocks of page structure. At least, this relates to the D*ocument* O*bject* M*odel* (DOM) based representation of a webpage's HTML structure and to its linguistic structure. As far as we deal with the latter, we concentrate on the notion of logical (text) document structure as introduced in Power et al. (2003). An XML-based framework for dealing with logical text document structure is the C*orpus* E*ncoding* S*tandard* (CES; Ide et al. 2000) which we integrate in part into our GXL-based model. The basic tenet for doing this is to have an integrated, encompassing representation of a webpage's internal structure.

DOM related information is extracted from the HTML source of the corresponding input page. In many cases this source cannot be parsed directly because of malformed code. We use the HTMLParser[5] for parsing and correction in order to overcome this

[5] `http://htmlparser.sourceforge.net`

```
<!DOCTYPE gxl SYSTEM "http://www.gupro.de/GXL/gxl-1.0.dtd">
<gxl>
    <graph hypergraph="true" edgemode="directed" id="HyperGraph0">
        <node id="ModuleA">
            <graph id="InternalLinkStructureA1" hypergraph="false" edgemode="directed">
                <node id="AnchorA1"><!--...--></node>
                <node id="AnchorA2"><!--...--></node>
                <node id="AnchorA3"><!--...--></node>
                <node id="AnchorA4"><!--...--></node>
                <!--...-->
                <edge id="EdgeA1" from="AnchorA2" to="AnchorA3">
                    <attr name="types"><set><string>internallink</string></set></attr>
                </edge>
                <!--...-->
            </graph>
        </node>
        <node id="ModuleB">
            <graph id="InternalLinkStructureB1" hypergraph="false" edgemode="directed">
                <node id="AnchorB1"><!--...--></node>
                <node id="AnchorB2"><!--...--></node>
                <!--...-->
            </graph>
        </node>
        <node id="ModuleC">
            <graph id="InternalLinkStructureC1" hypergraph="false" edgemode="directed">
                <node id="AnchorC1"><!--...--></node>
                <!--...-->
            </graph>
        </node>
        <rel id="Hyperlink1">
            <attr name="types"><set><string>kernellink</string></set></attr>
            <relend direction="in" target="ModuleA" role="sourcepage" startorder="1"/>
            <relend direction="in" target="AnchorA1" role="sourceanchor"/>
            <relend direction="out" target="ModuleB" role="targetpage" endorder="1"/>
        </rel>
        <rel id="Hyperlink2">
            <attr name="types"><set><string>downlink</string></set></attr>
            <relend direction="in" target="ModuleA" role="sourcepage" startorder="2"/>
            <relend direction="in" target="AnchorA4" role="sourceanchor"/>
            <relend direction="out" target="ModuleB" role="targetpage" endorder="2"/>
            <relend direction="out" target="AnchorB1" role="targetanchor"/>
        </rel>
        <rel id="Hyperlink3">
            <attr name="types"><set><string>kernellink</string></set></attr>
            <relend direction="in" target="ModuleB" role="sourcepage" startorder="1"/>
            <relend direction="in" target="AnchorB2" role="sourceanchor"/>
            <relend direction="out" target="ModuleC" role="targetpage" endorder="1"/>
        </rel>
    </graph>
</gxl>
```

Table 2. Schematic outline of a sample GXL-based representation of a website (dots indicate omitted content – note that in this and subsequent examples we use descriptive IDs which in runtime experiments are replaced by prefixed numbers)

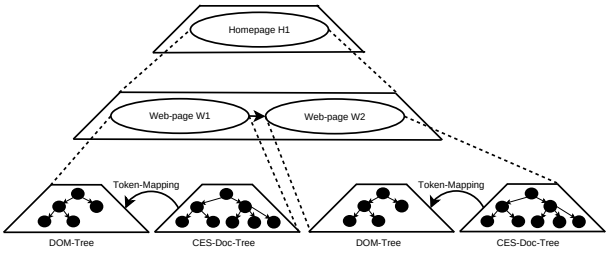

Figure 8. Integrated representation of DOM and LDS structure

problem. It provides an interface to the output DOM which GXL allows to represent as a directed rooted tree. We embed this tree into the node model of the focal page (see table 3 and figure 8). The DOM tree is the main source for deriving a page's internal link structure.

The second level of structure formation concerns the linguistic document structure of a webpage which we assume, for the sake of simplicity, to be representable as a labeled tree – this is, of course, an oversimplification, but serves as a working definition. We follow the approach of Power et al. (2003) and thus represent, amongst others, tokens, sentences, paragraphs and sections as part of a webpage's logical (text) document structure.

For various reasons, the extraction of this linguistic information from a webpage is not trivial. HTML possesses some basic means to represent document structure: for example, `H`-tags can be used to denote headlines and `P`-tags to mark paragraphs. But HTML lacks elements needed for explicitly tagging linguistic elements as, for example, sentences and tokens. Beside this insufficient expressiveness, another drawback is the *tag abuse* problem Barnard et al. 1995) which occurs when HTML tags are misused for layout purposes. Someone might, for example, use a headline tag to highlight a phrase in continuous text. On the other hand, the headline of a chapter could be highlighted by means of a bold

font without using a headline tag. Instead of going into the details of these problems when extracting document structure from DOM trees, we rather discuss the question of how to integrate the latter with representations of logical text document structure. In GXL, both structures can be represented as graphs. However, it would be insufficient not to account for their mapping. Because of differences in scope, we do not map their inner nodes or try to order or even to nest them, but rather focus on a mapping of their elementary text tokens only.[6] We do that by mapping each token of a page's text content model to the most specific node of the DOM tree to which it belongs. Figure 9 illustrates this mapping. In terms of a simplified GXL encoding, this example is outlined in table 3. The internal structure of `Module1` is represented by an additional embedded graph. This graph itself contains two embedded graphs which represent its DOM and logical document structure. Finally, the token-based mapping is manifested by a third graph.

So far, we have augmented our hypertext graph model by means of three component graphs which are nested into the nodes representing the pages whose link, DOM and linguistic structure they model. What is missing is an account of the fact that websites are hypertext documents which allow easy editing and modifications without necessarily losing their object identity. That is, we need to consider the revision process of (logically) the same website. This is outlined in the next section.

3.3 Time-aligned website representations

Web-based hypertexts are dynamic entities which preserve their "object identity" although they may change their gestalt dramatically during their lifespan. Above, the example of a conference

[6] It is easy to see that sentences may contain HTML-lists as list items can obviously contain sentences so that we cannot nest a webpage's logical document structure into its DOM structure nor the other way round.

```
<gxl>
    <graph hypergraph="true" edgemode="directed" id="HyperGraph0">
        <node id="Module1">
            <graph id="DOM-Tree1">
                <attr name="type"><enum>DOM</enum></attr>
                <node id="tag1"><!--...[body]...--></node>
                <node id="tag2"><!--...[h1]...--></node>
                <node id="html-text-1"><!--...[Conference 2005]...--></node>
                <edge from="tag1" to="tag2"/>
                <edge from="tag2" to="html-text-1"/>
                <!--...-->
            </graph>
            <graph id="CES-Doc1">
                <attr name="type"><enum>CES</enum></attr>
                <node id="node1"><!--...[body]...--></node>
                <node id="node2"><!--...[tok]...--></node>
                <node id="ces-orth1"><!--...[Conference]...--></node>
                <node id="ces-orth2"><!--...[2005]...--></node>
                <!--...-->
            </graph>
            <graph id="CES_DOM_Mapping1">
                <attr name="type"><enum>Mapping</enum></attr>
                <edge from="ces-orth1" to="html-text-1"/>
                <edge from="ces-orth2" to="html-text-1"/>
                <!--...-->
            </graph>
            <graph id="InternalLinkStructure1" hypergraph="false" edgemode="directed">
                <attr name="type"><enum>Linkage</enum></attr>
                <!--...-->
            </graph>
        </node>
        <!--...-->
    </graph>
</gxl>
```

Table 3. A sample nesting of webpage structure (dots indicate omitted content)

website was given, where its gestalt ranged from a single page at the time of its creation to possibly several hundred pages as the conference event approaches. At least, the following types of changes interrelating the interleaving website revisions can be distinguished when primarily focusing on webpages:[7]

- A `minor change` of a webpage typically concerns the correction of spelling mistakes or minor reformulations of its wording.
- A `significant change` of a webpage occurs when content

[7] The following listing does not claim to be a complete list of possible website changes.

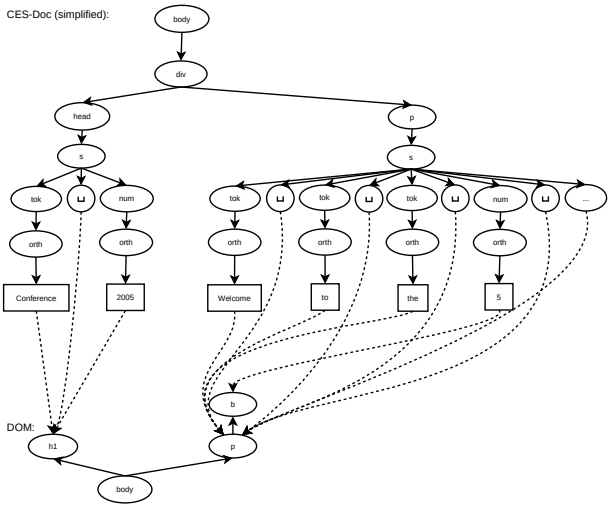

Figure 9. Mapping between text-tokens of DOM and CES representation

is added, removed or rearranged within the page.

- A `layout change` occurs when its layout is changed without actually touching its content.
- The deletion of a webpage is encoded as `deletion`. Analogously, the insertion of a webpage in a subsequent stage of a website's lifespan is encoded as `insertion`.
- A `replacement` of a webpage occurs if its content changes completely.
- The case of a `webpage movement` without replacement occurs if only the URL is changed.
- A `change of link structure` may have its source in webpages linking to the focal one. But also outgoing hyperlinks may have changed.

- The `movement of an entire home page or website` is a special case of a webpage movement. Regarding this type, it is assumed that the structure of the home page itself is not significantly affected.

In the previous sections, we have presented an integrated model of different levels of structure formation starting from a website's link structure down to the DOM structure of elementary pages. These representations are snapshots of Web-based hypertexts at certain points in time. In order to represent the order of these snapshots, we add a further representation layer on top of the existing ones. That is, we introduce a graph whose nodes denote website representations at certain points in time. The chronological ordering of these points in time is mapped by means of an additional directed graph.

The next step is to type the modifications that interrelate neighboring snapshots. We utilize the list of types of modifications presented above. If, for example, the content of a webpage has slightly changed, the respective website representations of the same website are interlinked by a `rel`(ation) of type `minor change`.[8] In the case of deletions and insertions simple `rel`s (i.e., hyperedges) each with only one `rel`(ation)`end` are used instead (see table 4).

This representation does, of course, not express the modification in detail, but it should be sufficient to quickly locate the places where changes occurred in order to analyze them separately. Figure 10 shows an example of a chronologically ordered representation of hypertext snapshots. This example can be encoded in GXL as outlined in table 4.

[8]The automatic detection of such changes is coming into reach by means of the framework of graph similarity measuring (Mehler et al. 2005).

```
<gxl>
    <graph id="snapshots_homepage_h1">
        <node id="snapshot_homepage_h1_2005-08-10">
            <graph id="document_network1">
                <node id="webpage_1_of_2005-08-10"/>
                <node id="webpage_2_of_2005-08-10"/>
                <!--...-->
            </graph>
        </node>
        <node id="snapshot_homepage_h1_2005-09-10">
            <graph id="document_network2">
                <node id="webpage_1_of_2005-09-10"/>
                <node id="webpage_2_of_2005-09-10"/>
                <!--...-->
            </graph>
        </node>
        <rel id="Hyperlink1">
            <attr name="types"><set><string>minor change</string></set></attr>
            <relend direction="in" target="webpage_1_of_2005-08-10" role="source"/>
            <relend direction="out" target="webpage_1_of_2005-09-10" role="target"/>
        </rel>
        <rel id="Hyperlink2">
            <attr name="types"><set><string>deletion</string></set></attr>
            <relend direction="in" target="webpage_2_of_2005-08-10" role="source"/>
        </rel>
        <rel id="Hyperlink3">
            <attr name="types"><set><string>insertion</string></set></attr>
            <relend direction="in" target="webpage_2_of_2005-09-10" role="source"/>
        </rel>
    </graph>
</gxl>
```

Table 4. Schematic outline of a GXL-based website representation (dots indicate omitted content)

4 Towards explorations of linguistic regularities sensitive to hypertext structure

Following the line of argumentation in Mehler (2005) and utilizing the representation model presented so far, we can now refer to website structure as a resource for (i) narrowing down the scope of linguistic pattern exploration and (ii) specifying additional constraints on those events which count as occurrences, co-occurrences, repetitions etc. In order to do that, the concept of a *domain* and, based on that, of a *data pool* have to be defined analogously to Mehler (2005).

In the present context, the notion of a domain is used to classify spans of the logical *hypertext* document structure of websites

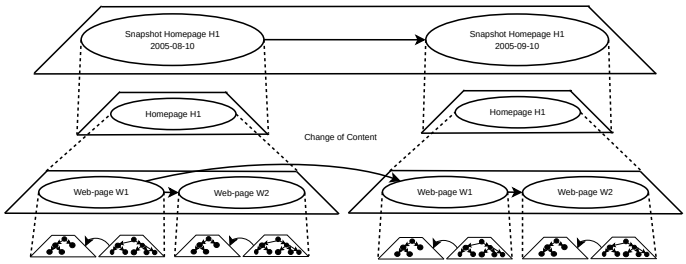

Figure 10. A time ordered website representation

and webpages as well as of the logical *text* document structure of the latter. Consequently, a domain equals, for example, a *module* type, a *Web document* type or a *document network* type. As we focus in this paper on expression units of Web-based communication, domains are seen to be additionally exemplified by the types *website*, *webpage* and all types of building blocks of the DOM and logical text document structure of webpages (e.g., *table*, *paragraph* and *sentence*). Finally, domains are seen to also include any type of spans as they are defined by parts of the kernel hierarchy and of the various levels of structure formation of single pages (e.g., *left subtree of the leader* or *third level of the right subtree of the leader* or *source and target page of an across link*). Thus, domains are used to type website spans in which linguistic data (e.g., co-occurrences) is observed.[9]

In the following definition, these types of spans of websites are referred to as elements d of the set of domains D. Further, if $S(C)$ is the set of all segments of the *webgenre document bank* C according to some segmentation procedure (segmenting, for example, websites into their pages and these pages into their sections, paragraphs and sentences etc.) and $s \in S(C)$ is a segment of type $d \in D$, then this is symbolized as $s \models_{S(C)} d$. If $\boldsymbol{a} \in T$ is a to-

[9]Note that we represent websites by means of GXL which is data-oriented and thus does not directly allow to specify domains using the XPath language.

ken (i.e., a (lexical) text position) instantiating (i.e., mapped onto the) type $a \in V$, we symbolize this as $\boldsymbol{a} \models_T a$. T and V are the set of tokens and types, respectively, so that $T(s)$ and $V(s)$ are analogously the set of tokens and types of the segment s. We now redefine definition 1 of Mehler (2005) and extend it in order to make it applicable to websites:

Definition 1. Let C be a webgenre document bank and $d \in D$ a domain with the set of instances $d(C) = \{s \in S(C) \,|\, s \models_{S(C)} d\}$ in C. The set of all co-occurrences of any types in segments of the domain d is $\Omega_C^d = \{(\boldsymbol{a}, \boldsymbol{b}) \,|\, \exists s \in d(C) : \boldsymbol{a}, \boldsymbol{b} \in T(s) \wedge \boldsymbol{a} \trianglelefteq \boldsymbol{b}\}$. The relation $\trianglelefteq$ maps the syntagmatic order of the textual content of the elements (i.e., websites) of C. On the level of websites, this order is based on the depth first order of their component pages according to the kernel hierarchy. On the level of webpages, it is based on the syntagmatic order of their text content. $\boldsymbol{a} \trianglelefteq \boldsymbol{b}$ means that $\boldsymbol{a}$ is a text position (i.e., a token) which linearly occurs before text position $\boldsymbol{b}$. With the help of Ω_C^d several sets can be derived:

1. $\Omega_C^d|_{(a,b)} = \{(\boldsymbol{a}, \boldsymbol{b}) \in \Omega_C^d \,|\, \boldsymbol{a} \models_T a \wedge \boldsymbol{b} \models_T b\}$ is the set of all co-occurrences of $a, b \in V$ in segments of the domain d, in which a occurs before b.

2. $\Omega_C^d|_{\{a,b\}} = \Omega_C^d|_{(a,b)} \cup \Omega_C^d|_{(b,a)}$ is the set of all co-occurrences of $a, b \in V$ in segments of domain d irrespective of their syntagmatic order.

3. $\Omega_C^d|^x = \{(\boldsymbol{a}, \boldsymbol{b}) \in \Omega_C^d \,|\, \boldsymbol{a}, \boldsymbol{b} \in T(x)\}$ is the set of all co-occurrences of any types in segment x of the domain d.

4. $\Omega_C^d|^x_{(a,b)} = \{(\boldsymbol{a}, \boldsymbol{b}) \in \Omega|_{(a,b)} \,|\, \boldsymbol{a}, \boldsymbol{b} \in T(x)\}$ is the restriction of $\Omega_C^d|_{(a,b)}$ to x. Accordingly, $\Omega_C^d|^x_{\{a,b\}} = \Omega_C^d|^x_{(a,b)} \cup \Omega_C^d|^x_{(b,a)}$.

5. $h_{ij} = |\,\{\boldsymbol{a} \,|\, \exists (\boldsymbol{b}, \boldsymbol{c}) \in \Omega_C^d|^{x_j} : \boldsymbol{a} \models_T a \wedge (\boldsymbol{a} = \boldsymbol{b} \vee \boldsymbol{a} = \boldsymbol{c})\}\,|$ is the frequency of $a_i \in V$ in segment x_j of domain d.

Ω_C^d and any set derived from it according to the latter specifications is called *data pool induced by* the corpus C, the domain d and possibly

some additional restrictions separated by |. □

Definition 1 is easily extended by additional co-occurrence restrictions. The restriction which is mostly applied in this context is the frequency restriction used to rule out *hapax legomena* and other low frequency items. Another frequently used restriction refers to the syntagmatic distance of the units to be viewed as co-occurring. These and related restrictions are not formalized in the present paper – we leave that to future work.

Data pools according to definition 1 work as filters which make accessible the linguistic information of Web-based communication as it is distributed over websites. The aim is to make it accessible to the various tasks of exploratory corpus analysis and machine learning by preserving restrictions as they result from the possibly genre-specific structuring of websites. Following this line of argumentation, co-occurrence analyses, for example, no longer need to be restricted to the textual content of *single* pages, but may include co-occurrences of items belonging to different but neighboring pages of the same level of the kernel hierarchy. To give another example: Co-occurrence analyses may be solely based on pages which are interlinked by means of across links. As websites are characterized by the phenomenon of discontinuous manifestation (see section 2) and related aspects of informational uncertainty, such an approach is indispensable when analyzing dependencies of linguistic items which, though they deal with the same topic or manifest the same function, are nevertheless distributed over different pages. This is exemplified by a conference website (e.g., `http://www.ht04.org/`) in which the (conference) program section is distributed over several webpages (i.e., three pages in the case of the latter example) so that there is, for example, no co-occurrence on any of these pages of the types *paper* and *keynote* (except for the menu). The aim of definition 1 is thus to soften or even neutralize such limitations in a way which is grounded in the underlying webgenre structure model.

5 Conclusion

In this paper, we have presented a GXL-based model for the representation of the link structure of websites, the nested structure of their constitutive pages and the alignment of their successive snapshots. This was proposed as a preliminary step to automatically analyzing and representing webgenres as they are instantiated by websites. In Mehler and Gleim (2005), the distribution of hypertext graphs of the genre of conference websites is analyzed based on this framework. In Mehler et al. (2005), the present framework is utilized to derive an algorithm for unsupervised graph learning. In this paper it is demonstrated that the link and DOM structure of websites and pages, respectively, are valuable sources for hypertext categorization. Improvements in this area hinge on improving hypertext representation. As has been shown, this task poses a lot of problems which, we believe, can only be adequately solved by means of machine learning methods *grounded in a webgenre model*. Future work will address these induction methods and their grounding in more detail. Analogously to the algorithm proposed in Mehler et al. (2005), these methods will be settled in the framework of unsupervised graph learning.

References

Amitay, E., Carmel, D., Darlow, A., Lempel, R. and Soffer, A. (2003).The connectivity sonar: Detecting site functionality by structural patterns. *Proceedings of the 14th ACM conference on Hypertext and Hypermedia*, 38-47.

Barnard, D. T., Burnard, L., DeRose, S. J., Durand, D. G. and Sperberg-McQueen, C. (1995). Lessons for the World Wide Web from the text encoding initiative. *Proceedings of the Fourth International WWW Conference "The Web Revolution"*.

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. *Proceedings of LREC 2004*, 1313-1316.

Biber, D. (1995). *Dimensions of register variation: A cross-linguistic comparison*, Cambridge: CUP.

Botafogo, R. A., Rivlin, E. and Shneiderman, B. (1992). Structural analysis of hypertexts: Identifying hierarchies and useful metrics. *ACM Transactions on Information Systems* 10(2), 142-180.

Chakrabarti, S. (2002). *Mining the Web: Discovering knowledge from hypertext data*, San Francisco: Morgan Kaufmann.

Crowston, K. and Kwasnik, B. (2003). Can document-genre metadata improve information access to large digital collections? *Library Trends*.

Crowston, K. and Williams, M. (1999). The effects of linking on genres of Web documents. *Proceedings of the Hawai'i International Conference on System Science*.

Crowston, K. and Williams, M. (2000). Reproduced and emergent genres of communication on the World Wide Web. *The Information Society* 16(3), 201-216.

Dillon, A. and Gushrowski, B. (2000). Genres and the Web: Is the personal home page the first uniquely digital genre? *Journal of the American Society of Information Science* 51(2), 202-205.

Eggins, S. (1994). *An introduction to systemic functional linguistics*, London: Continuum.

Eiron, N. and McCurley, K. (2003). Untangling compound documents on the Web. *Proceedings of the 14th ACM Conference on Hypertext and Hypermedia*, 85-94.

Firth, D. and Lawrence, C. (2003). Genre analysis in information systems research. *Journal of Information Technology Theory and Application* 5(3), 63-87.

Furner, J., Ellis, D. and Willett, P. (1996). The representation and comparison of hypertext structures using graphs. In Agosti, M. and Smeaton, A. (eds.) *Information Retrieval and Hypertext*, 75-96.

Getoor, L. (2003). Link mining: A new data mining challenge. *SIGKDD Explorations Newsletter* 5(1), 84-89.

Gleim, R. (2005). HyGraph: Ein Framework zur Extraktion, Repräsentation und Analyse webbasierter Hypertexte. In Fisseni, B., Schmitz, H., Schröder, B. and Wagner, P. (eds.) *Sprachtechnologie, mobile Kommunikation und linguistische Ressourcen*, 42-53.

Halliday, M. and Hasan, R. (1989). *Language, context, and text: Aspects of language in a socialsemiotic perspective*, Oxford: OUP.

Heinemann, W. (2000). Textsorte – Textmuster – Texttyp. In Brinker, K., Antos, G., Heinemann, W. and Sager, S. F. (eds.) *Text- und Gesprächslinguistik. Linguistics of text and conversation*, Berlin: de Gruyter, 507-523.

Ide, N., Bonhomme, P. and Romary, L. (2000). Xces: An XML-based standard for linguistic corpora. *Proceedings of LREC 2002*, 825-830.

Jakobs, E. (2003). Hypertextsorten. *Zeitschrift für germanistische Linguistik* 31(2), 232-252.

Keller, F. and Lapata, M. (2003). Using the Web to obtain frequencies for unseen bigrams. *Computational Linguistics* 29(3), 459-484.

Kilgarriff, A. and Grefenstette, G. (2003). Introduction to the special issue on the Web as corpus. *Computational Linguistics* 29(3), 333-347.

Martin, J. (1992). *English text. System and structure*, Amsterdam: Benjamins.

Mehler, A. (2005). Preliminaries to an algebraic treatment of lexical associations. *Proceedings of the Workshop Learning and Extending Lexical Ontologies at ICML 2005.*

Mehler, A., Dehmer, M. and Gleim, R. (2004). Towards logical hypertext structure – a graph-theoretic perspective. *Proceedings of I2CS '04*, 136-150.

Mehler, A. and Gleim, R. (2005). Polymorphism in generic Web units. A corpus linguistic study. *Proceedings of Corpus Linguistics 2005*, available online at `http://www.corpus.bham.ac.uk/PCLC/`.

Mehler, A., Gleim, R. and Dehmer, M. (2005). Towards structure-sensitive hypertext categorization. *Proceedings of the 29th Annual Conference of the German Classification Society.*

Melnikov, O., Tyshkevich, R., Yemelichev, V. and Sarvanov, V. (1994). *Lectures on graph theory*, Mannheim: BI Wissenschaft.

Miller, G. and Charles, W. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes* 6(1), 1-28.

Orlikowski, W. and Yates, J. (1994). Genre repertoire: The structuring of communicative practices in organizations. *Administrative Science Quarterly* 39(4), 541-574.

Power, R., Scott, D. and Bouayad-Agha, N. (2003). Document structure. *Computational Linguistics* 29(2), 211-260.

Rehm, G. (2002). Towards automatic Web genre identification: A corpus-based approach in the domain of academia by example of the academic's personal homepage. *Proceedings of the Hawai'i International Conference on System Sciences*.

Resnik, P. and Smith, N. (2003). The Web as a parallel corpus. *Computational Linguistics* 29(3), 349-380.

Routledge, L., Bailey, B., van Ossenbruggen, J., Hardman, L. and Geurts, J. (2000). Generating presentation constraints from rhetorical structure. *Proceedings of the 11th ACM Conference on Hypertext and Hypermedia*, 19-28.

Santamaría, C., Gonzalo, J. and Verdejo, F. (2003). Automatic association of Web directories to word senses. *Computational Linguistics* 29(3), 485-502.

Storrer, A. (2002). Coherence in text and hypertext. *Document Design* 3(2), 156-168.

van Dijk, T. and Kintsch, W. (1983). *Strategies of discourse comprehension*, New York: Academic Press.

Ventola, E. (1987). *The Structure of social interaction: A systemic approach to the semiotics of service encounters*, London: Pinter.

Winter, A., Kullbach, B. and Riedinger, V. (2002). An overview of the GXL graph exchange language. In Diehl, S. (ed.) *Software Visualization*, 324-336.

Yates, J. and Orlikowski, W. (1992). Genres of organizational communication: A structurational approach to studying communications and media. *Academy of Management Review* 17(2), 299-326.

Yoshioka, T. and Herman, G. (2000). Coordinating information using genres. Technical Report, MIT Sloan School of Management.