Humanities

Koenraad de Smedt[1] and William J. Black[2]

[1] Section of Linguistic Studies
University of Bergen, Norway
Email: desmedt@uib.no

[2] Department of Language Engineering
University of Science and Technology, Manchester
Email: bill@ccl.umist.ac.uk

Summary. Innovation in humanities education and research is stimulated by new technologies for the processing of language, speech, music, visual arts, and other expressions of the human mind. Three main topics will be discussed: the role of large scale resources such as text corpora and digital archives; advanced methods and tools for processing and simulation; and the use of courseware, multimedia, and hypermedia in humanities teaching and learning.

1 Introduction

The humanities study products of the human imagination, as represented in a wide range of symbolic (mostly linguistic), auditory, and visual forms. As computing during the second half of the 20th century gradually shifted from numeric applications to multimedia, the new possibilities showed perhaps as strongly in the humanities as in anywhere. By the beginning of the 1990s, word, picture, and sound processing had come within easy reach of nearly all faculty and students. This has opened up enormous potential for new forms of learning and teaching where the very objects of study of humanities scholars could be brought to everyone's computer almost without limitations.

Of all expressions of human culture and the mind, written language is the easiest to represent digitally, because writing is essentially based on discrete, symbolic codes. Already in the late 1940s, father Roberto Busa SJ conceived the plan to make a large concordance of the works of Thomas Aquinas. When presented with the plan, people at IBM thought it would be impossible to make such an index for all these works, but Busa persisted. This was not the last time a humanities scholar pressed the state of the art beyond its limits. Today, researchers in the humanities are contributing to many important innovations in information technology, such as text encoding, visual search in pictorial databases, machine translation, and speech understanding. Much progress has been made in the last half of the 20th century and there is no area of humanities education and research that has not been affected by information processing methods.

While the state of the art in computer technology has moved from punched cards to colorful screen images, the basic objects of study of the humanities have remained largely the same. Thus, the various humanities disciplines study, for instance, the complexity of human language, the laws governing logical thought, the genius of artistic creation, the interpretation of history, the rhetoric of speech, the definition of literary styles, the ubiquity of religion, or the cultural expressions in our societies. However, the methods for studying these may vary depending on the objects of study and the available technology.

Exactly which disciplines are included in the humanities varies somewhat depending on national and cultural setting. At many universities, linguistics is part of the humanities, while others rank it among the social sciences. *Philosophy* ranks among the humanities at many universities, but some treat it as separate, and even assign it a separate faculty. In the current chapter, we will take a broad, inclusive standpoint and assume that the humanities (or liberal arts or human sciences) may include *literary scholarship*, *languages* and *linguistic studies*, *history*, *history of art*, *architecture* and *design*, philosophy and *logic*, *aesthetics*, *religion*, *music*, *archaeology*, *cultural studies*, and *philosophical anthropology*. This list is not meant to exclude other fields and does not exclude the possibility that some research and teaching relevant to the humanities can be carried out in other departments and faculties, such as psychology, social sciences, and computer science.

Providing students with computers is a necessary first step towards a renewal of humanities education in the information age, but it is a grossly insufficient one. Reading a book on-line rather than in the old library, or writing an essay on-line rather than on paper, will hardly make students better prepared for the information society. Three new profiles of humanities graduates are emerging, with the following desired competencies:

1. graduates who can integrate new computing methods in traditional humanities jobs: the archive, library, museum, publishing, culture, and entertainment sectors;
2. graduates who can integrate humanities knowledge in traditional information technology jobs: the communication industries with applications in telecommunications, media, car manufacturing, distance education, etc.;
3. graduates who can develop new humanities information technology: research for human language industries, information retrieval, computer games, digital art, text coding, etc.

In order to provide these competencies, new methods based on information technology must be acquired by humanities students. On the one hand, experience has shown that it is pointless to try and teach computing to a humanities student without integrating it into the student's domain of expertise. On the other hand, it is equally pointless to try and teach computing methods in any discipline without an understanding of what computing is all about. A careful balance must therefore be achieved between the teaching of computer science concepts and their application in scientific, discipline-oriented methods.

In linguistic studies, such methods may for instance include formal systems for the automatic syntactic analysis of sentences. In music studies, new methods may involve acoustic filtering, or quantization of musical rhythms. In literary and textual studies, some new methods are based on a statistical analysis of texts in order to help determine their authorship or categorize them objectively with respect to style. Art historians may need to use image recognition algorithms, and so on. The most fundamental characteristic which these methods have in common is that they are formal: the object of study is encoded as formal data to which formally specified algorithms may be applied. Also, in practice, these methods cannot be learned effectively from books, but need hands-on experience.

Unfortunately, it is a common misconception that learning to use new technologies in the humanities can be achieved by the mere introduction of multimedia in the classroom. By itself, multimedia merely acts as a gift wrapper around old, static content that leaves little room for active manipulation by students. The interaction provided by typical so-called *interactive* multimedia systems is often very limited in nature, and puts the student in a closed system with predefined pieces of information and predefined answers to questions. New competencies can be achieved only by letting humanities students actively experiment with scientific computing methods applicable to their specific field.

During the 1980s and the early 1990s, the possibilities of the new technologies for the humanities became more widely recognized, particularly when the significance of the personal computer revolution started to become apparent. It was in this period that the potential for computing to transform teaching and learning as well as research in the text-based disciplines began to receive attention, and courses and programs formally involving the application of computing were initiated in a number of European institutions. In some countries this was accompanied by national initiatives, such as *Humanistisk Datasenter* in Norway, or the *Computers in Teaching Initiative* (`http://www.cti.ac.uk/`) in the UK, which set up two centers relevant to the text-based disciplines: the CTI Centre for Textual Studies at Oxford, and the CTI Centre for Modern Languages at Hull.

National and local initiatives have played a supporting role in the creation of new courses and degrees in humanities computing. Given that the different disciplines each have their own objects of study and their own traditional methods, one may wonder if there is a single canon of humanities computing methods that justifies dedicated humanities computing courses or degrees. On the one hand, the specific methods applied in the various humanities disciplines may be too varied and specialized to be covered in a common course. On the other hand, some commonalities may be found, notably with respect to formal methods, statistics, textual scholarship, and general research methods, that cuts sufficiently across many disciplines to justify a dedicated course.

Over the past decades, many specializations combining the new technologies with traditional humanities disciplines have achieved varying degrees of success. Of these, computational linguistics is probably the most well-established and has gained recognition as a separate discipline, combining a rich tradition of well-known methods with continuous innovation. Meanwhile, other new fields, such as historical informatics, are finding their role but are less perceived as separate disciplines. Furthermore, dedicated humanities computing curricula and degrees are trying to position themselves in various ways. While some humanities computing courses explore fundamental methodological questions in text encoding, others are instrumental in the practical teaching of how to use tools, and still others try to find niches rarely covered by other humanities disciplines, such as computer games.

In the remainder of this section, we will make reference both to humanities computing methods that cover a wide range of disciplines and to some discipline-specific approaches, in an attempt to cover at least some balance between breadth and depth in the limited space available. The range of examples is, of course, far from exhaustive. First, we focus on the use of large-scale computing resources, such as corpora and archives. Then, we shift attention to computer methods and tools for processing language and other materials. Finally, we discuss more generally the role of courseware, multimedia and hypermedia in humanities education.

2 Large-scale resources

If there is one thing nearly all humanities disciplines have in common, it is the need to study large collections of texts, sounds, and pictures. Traditionally, indeed, humanities students spent much time in libraries, archives, and museums. Even if the same resources may be studied in different disciplines, scholars may have different research goals and take rather different perspectives. A lexicographer may wish to count the occurrence of words and their meanings in a collection of texts, while a scholar of literature may study the use of style in the same collection of texts, and the historian may want to extract historical facts or interpretations from it.

Increasingly, the essential reference works used by students are available in electronic form, either online via WWW or on CD-ROM, while a possibly even larger collection of materials is not necessarily intended for formal learning but can sometimes be used for for that purpose. Large text corpora make it possible to carry out new kinds of analyses that take into account all the works of an author or all the works of a particular location, period, or genre. In comparison with printed reference works, electronic versions of texts for the humanities can be encoded and made searchable, indexed or cross-referenced for different purposes (e.g., literary, historic, linguistic, cultural, or religious), which enlarges the scope of interests, and encourages interdisciplinary approaches.

Dictionaries lend themselves to electronic access, especially when they provide links to related meanings, as in a thesaurus. An early out of copyright edition of Roget's thesaurus is available electronically, but a more structured semantic dictionary is the *WordNet* (`http://www.cogsci.princeton.edu/~wn/`), developed at Princeton by George Miller and associates (Fellbaum, 1998). The user of this dictionary can find words whose senses are more specific (hyponyms) or general (hypernyms), as well as words related in other ways (part-whole or *meronymy* relations, antonymy, etc.). It has a simple user interface, illustrated in Fig. 1, but can also be used from the command line, or accessed by programs, and it runs on PCs, Macintosh, and Unix computers. More recently, a large scale European project has given rise to the *EuroWordNet* (`http://www.hum.uva.nl/~ewn/`), which lets the user explore how meanings can be expressed in several European languages.

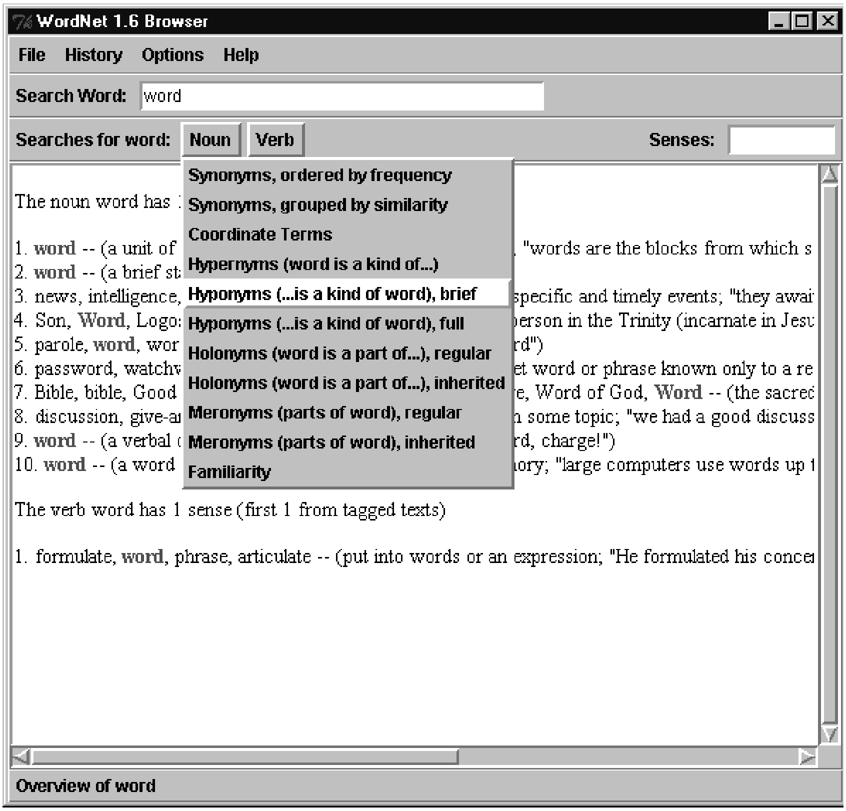

Fig. 1. The WordNet semantic dictionary's user interface, showing the menu of semantic relations that can be explored.

Electronic dictionaries and other lexical materials are not a primary source of information for the student of the humanities, but are themselves produced as the result of textual scholarship involving the study of source materials, original texts, and in some cases transcribed speech.

In general, textual, auditory, and pictorial material must be carefully encoded in order to be useful material for study purposes. The computer is at once more limited and more powerful than the book. It is more limited, because common computer screens appear coarser than the printed page and do not provide the same physical appearance and visual context. However, with proper treatment, computer-coded materials can be dynamically linked in various ways not possible in print, for instance by making detailed cross-references from a text edition to the original manuscript. Computer representations also allow different dynamic views of the same material, including color filtering, magnification, different angles, superposition, etc. Last but not least, computers allow many ways of indexing and fast searching.

As a first example of computer-coded material, consider *The Wittgenstein Archives* (`http://www.hit.uib.no/wab/`) at the University of Bergen. The 20,000 hand-written pages which Wittgenstein left after his death have been converted into machine-readable form and made available through a publication on CD-ROM in cooperation with Oxford University Press. In order to appreciate the added value offered to students by means of this electronic edition, in comparison to a printed edition, the numerous problems of the material must be understood. Since Wittgenstein himself never prepared more than a fraction of his writings for publication, most of the manuscripts and typescripts are full of annotations, deletions, insertions, marginal remarks, critical instructions, and cross-references, alternative formulations for particular phrases, and even writings in secret code (*Geheimschrift*). Neither is it always clear which of such alternative formulations he finally decided upon.

To reproduce the texts as completely as possible, The Wittgenstein Archives have developed their own text coding system MECS (Multi-Element Code System), which provides the basis for specially designed software that offers wide flexibility in the presentation and analysis of the texts. The text coding allows various linked presentation formats (from facsimile to diplomatic to normalized), code extraction (such as references to persons), variant control, word lists (per language, alphabetical, or frequency-based), and other statistical data. Summing up, the electronic version offers presentation and search possibilities for studying Wittgenstein's literary estate that would be hard to obtain in a printed edition. In general, indeed, electronic editions offer students of philosophy, literature, history, linguistics, and other disciplines a clearly more powerful way of accessing texts for study than print does.

Another, quite different example of a computer-coded resource is the *National Digital Archive of Norway* `http://www.hist.uib.no/arkivverket/`. This is a fully digitized archive which contains material from the Norwegian censuses. If properly coded and made accessible, digital archives such as this one can be very valuable for scholarship from historical, ethnological, and other perspectives. Some of the original material for the *National Digital Archive of Norway* (`http://www.hist.uib.no/arkivverket/`) is centuries old and required careful encoding in order to extract useful statistics. With the simple WWW interface, scholars of history or religion can, for instance, map occupation against marital status, to find out whether in 1801, a higher percentage of clergy were married than of craftsmen, or vice versa. Such exercises with real materials are much more rewarding for students than reading the same facts in a book, and have better pedagogical value.

A further example is the *European Manuscript Server Initiative* (EMSI, `http://linux2.hit.uib.no/vol/emsi/emsi.html`) which aims to provide electronic access to medieval and early modern manuscripts of Europe's libraries. These manuscripts are among the most significant parts of our cultural heritage and are relevant for a wide range of humanities disciplines, including scholars of literature, linguistics, history, ethnology, religion, gender studies, and cultural studies. Access to the physical manuscripts was, and still is, restricted to a handful of researchers, but EMSI wants to make the materials available in high-quality electronic form - better than conventional print - to all students.

Making manuscripts accessible to the public involves a range of problems. Technical ones include for instance color calibration. Others include a presentation structure that reflects the original organization of the manuscript in a multi-media environment. One must also address the concerns of the institutions holding the manuscripts regarding publishing rights. These institutions are often dependent on revenues from permissions to create facsimiles. In the world of digital media, it is often hard to find a clear-cut distinction between highly profitable usage and non-profit usage of some material. It is clear that easy access would allow the creation of improved teaching materials for university teaching throughout Europe, which, owing to the relatively specialized market, would provide only slim, if any margins of profit. In some cases, however, the same materials could be used to address secondary education, with a much larger market and, therefore, a bigger profit margin.

Also in linguistics, the use of large textual resources is greatly facilitated by computer techniques. *Corpus linguistics* is concerned with automatic methods for the *empirical* study of language from texts and speech. It has in common with other humanities disciplines that it relies on large quantities of textual or speech material. But whereas students of literature and history use a text corpus mainly for content and style, students of linguistics use it to study the occurrence and context of words, syntactic constructions, or other linguistic elements. Clearly, this requires texts to be coded quite differently.

As an example, consider the *British National Corpus* (BNC, `http://info.ox.ac.uk/bnc/`), a one hundred million word corpus of written and spoken British English, which can be consulted via the WWW. If a student wants to find patterns of certain English words, for instance, the words can simply be looked up and will be presented each in the line of text in which they occur. It is crucial, however, to be able to restrict searches to certain parts of speech. Thus, one may want to search for *lead* as a verb versus *lead* as a noun. For this purposes, all words in the BNC, as in most other linguistic corpora, are marked up with their parts of speech. The SARA corpus browser supports several types of query: individual words and phrase patterns, with or without part of speech tags, for example.

Corpora are not limited to written form. The *Bergen Corpus of London Teenage Language* (COLT, `http://www.hit.uib.no/colt/`) is an example of a corpus of spoken language where speech recordings are coupled to their transcriptions. This enables students to search for written strings, for instance *init*, and hear the speech fragments in which the given phrase occurs. Both spoken and written corpora must be carefully encoded. The BNC is encoded according to the guidelines of the *Text Encoding Initiative* (TEI, `http://etext.virginia.edu/TEI.html`), an international consortium aiming at standardization of text encoding that satisfies scholarly needs.

Through the availability of carefully coded and searchable materials, the boundaries between *research* and *teaching* is narrowing, with consequent benefits from a pedagogical perspective. Increasingly it is possible for students to work with source materials in digitized form, and to use the same reference works as researchers. Moreover, the electronic nature of the material encourages active exploration through the use of tools, which is a form of learning by doing. Allowing students to search actively in electronic resources has some obvious pedagogical advantages over offering them static facts in books. Also, the integration of different types of materials, e.g., texts and images, enlarges the scope of interest of the textual scholar and encourages broader disciplinary views.

Although the use of such large-scale resources in humanities education can offer great benefits to students, its effective deployment is at present hindered by many difficulties. On the one hand, material that originated from academic research projects is often free but difficult to access, due to lack of dissemination efforts in academic research and lack of user-friendly interfaces. On the other hand, material which is commercially distributed is easy to order and access, but often at prohibitive prices. At the technical level, much of the material, especially the commercial material, is readable and to some extent searchable, but cannot be processed in new, student-defined ways. At an organizational level, there is a lack of teaching staff trained in the use of such resources and a lack of technical support staff.

Furthermore, the accessibility of electronic texts and other data to humanities students crucially depends on adequate text encoding and mark-up. Relatively early in the history of modern computing, the set of 26 latin characters used in the English language was encoded according to standards such as 7-bit ASCII and EBCDIC, which today are regarded as inadequate. Recording the structure and appearance of a source text can, even at the basic level, be difficult for languages with scripts different from the Latin script. Not only are languages such as Arabic, Hebrew, and Persian written from right to left, but they do not mark vowels or mark them defective, such that, for some educational purposes, vowels must be painstakingly added into text with consonant characters only. Vietnamese uses diacritics both to designate special consonants and vowels, but also to designate tone. For some characters there will therefore be two diacritics, which must be placed in a special configuration with respect to each other, sometimes dependent on context.

Text mark-up also plays a crucial mediating role towards more sophisticated analytical approaches, not only by encoding parts of speech to assist study of an author's use of language, but by making semantic or other encodings to enable more overtly interpretative analyses. The work of the *Text Encoding Initiative* (TEI, `http://www.tei-c.org/`), funded in part by the European Commission, has played a significant role in the development of this work, ensuring that efforts that go into mark-up can be re-used by other scholars and preserved for the future. By adopting SGML and its offspring XML as the basis for its recommendations, the TEI ensured that scholars would be able to take advantage of software tools developed for the commercial world.

National or local centers for the humanities often function as repositories of data. They either hold electronic texts and other data themselves, or provide access to reliable or verified resources. By storing and providing data, these centers play a vital role in making important study materials available to students and researchers.

Moreover, repositories and humanities research centers also often function as centers for the teaching of advanced methodologies. For more information on the issue of *Digital Libraries* see chapter 38.

3 Tools for processing and simulation

Many computer tools used in education were not initially designed for teaching and learning purposes. Some tools are general, widespread tools, like the visual filters in photographic retouching programs that Jerome McGann found to be useful in revealing underlying patterns in digital reproductions of paintings.

But also tools that were specifically developed for research purposes in the humanities have been transferred for use in learning materials. In general, the use of computer tools encourages students to actively explore of the subject matter more than reading does. Nevertheless, all tools should be used with methodological caveats; tools may turn out to be inadequate because their underlying assumptions are inadequate. A few tools will be discussed in two fields: computational linguistics and logic.

3.1 Computational linguistics

Academics in *computational linguistics* (CL) are generally concerned that one aspect of their discipline that distinguishes it from general linguistics is the emphasis on practically feasible processes of analysis and generation by computer. The ability to do computational linguistics is at least as important a learning objective as learning about computational linguistics. In a survey on computational linguistics in Europe (ACO*HUM CL Survey 1999), the emphasis on group and individual exercises and projects bears this out. The results of this survey show that the profile of teaching methods used is much more like that for other technology disciplines than for traditional humanities teaching, with 37.4% of teaching based on exercises and projects. Therefore it is not surprising that the development of educational materials in this discipline has a focus on practical tools and technologies used to support the learning of how to do computational linguistics.

In CL courses and computer programs, multimedia is seldom used gratuitously. Generally, CL teaching methods use advanced simulation environments incorporating algorithms for the natural language processing methods that are the object of teaching. In these environments, the student can experiment freely and is not restricted to some predetermined answer to a problem. Such tools are necessarily based on the results of scientific and technological research in CL, although not necessarily the state of the art.

The core topics in CL, for most centers, are *parsing algorithms* (programs for analyzing sentences), *formal grammars*, representation of *lexical knowledge*, *formal semantics*, state of the art *grammar formalisms*, mathematics and logic, and pragmatic techniques. These are all areas in which learning how to apply the techniques is inseparable from learning about them. The usual manner in which these techniques are learned is via the use of computational mechanisms which implement the formalisms. Thus, techniques of formal grammars are explored through the writing of grammar rules which are subsequently fed into parser that attempts to analyze sentences with the help of the given grammar.

Learning how to assign structures to sentences of a natural language is a skill that should be expected of general linguists as well as computational linguists. This skill is strongly supported by systems for the automatic analysis (parsing) or generation of sentences. Examples include the *LFG Grammar Writer's Workbench for Lexical Functional Grammar* (`http://www.parc.xerox.com/istl/groups/nltt/medley/`, Fig. 2) and its successor the *XLE* platform (`http://www.parc.xerox.com/istl/groups/nltt/xle/`), both developed by Xerox Parc) and the *Grammar Laboratories for the Macintosh* by Linguistic Systems. Such tools not only perform sentence analysis, but also present the sentence structures visually in graphical ways which linguists are used to, including hierarchical tree structures and feature-value matrices. The educational benefits are considerable compared with the more usual pedagogy of, e.g., definite clause grammar, which needlessly brings a programming language into the picture, but not in a way that students are enabled to develop programming skills. The visualization of the output of analysis is already a significant benefit compared with textual output.

The LFG Grammar Writer's Workbench and its successor, the XLE platform, are being used for the teaching of syntax to beginning students if linguistics and computational linguistics, for instance at the University of Bergen, where they use it for writing syntactic rules for Norwegian, and find it useful to get immediate feedback on how the rules relate a sentence to its syntactic structure. As an example Fig 2 shows a grammar rule window, a lexicon window, and two windows with different representations of the sentence structure: a tree structure (top right) and a feature structure (bottom right). There are plans to use it also in their study of their non-Indo-European language. Although the study of such a language is a popular part of the linguistics program, students have often complained that there is too little connection between the study of this language and the linguistic theories they learn about. Actually writing syntactic rules for various grammatical constructions in the non-Indo-European language and using the workbench to test whether their rules correctly analyse these constructions, would provide students with a much more concrete link between these two parts of the study program.

Writing grammars is of course a part of what students in linguistics, CL, and even languages need to learn, and probably a major part. But computational linguists also need to understand how the algorithms for processing linguistic data actually work. Few of the tools available provide an insight into the process of linguistic analysis or generation. Typically, the system operates as a black box between reading the user's input and presenting the results of analysis. It does not show the process of analysis, something that can be of particular learning value to a student.

If, in addition to *using* a parsing tool, CL students must learn to understand *how* such a tool works, techniques like visual stepping or animation can be useful. For example, a visual representation of the step-by-step growing search space

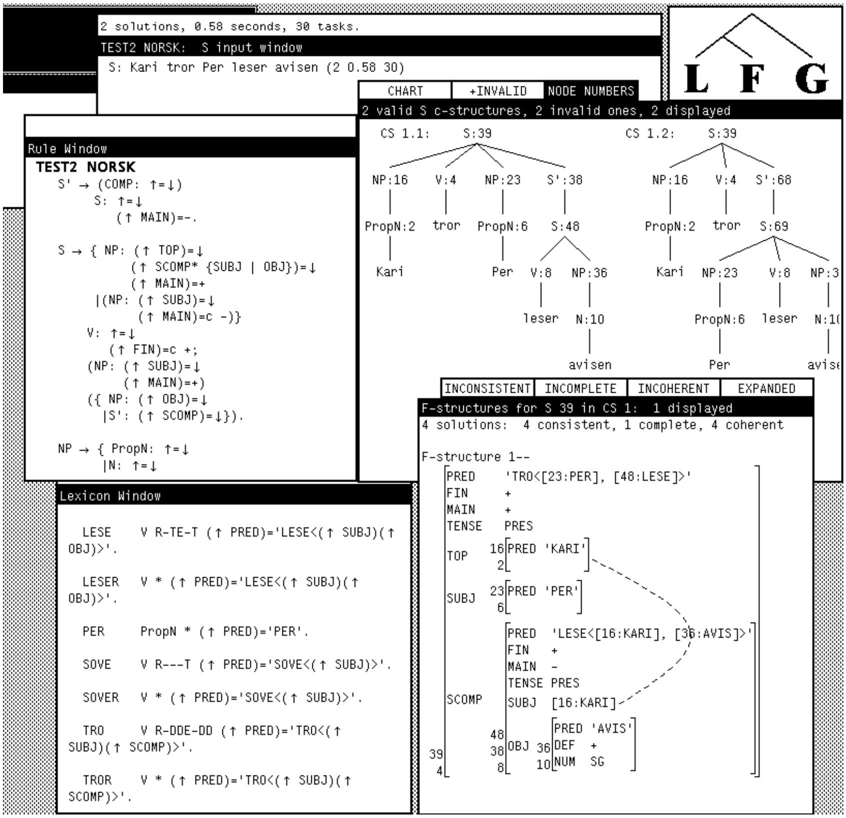

Fig. 2. Screen shot of the LFG Workbench

of a top-down parser referring to a left-recursive rule can help the student understand the limitations of that particular parsing mechanism. The *PAIL* laboratories (`http://www.idsia.ch/pail.html`) developed at IDSIA included a range of different parsing algorithms which could be explored in this way. A very simple tool that helps the student understand the mechanics of deriving a phrase structure analysis is described in Black, Hill, and Kassaei (1999). In this tool, the student enters a string, and then manually selects production rules one at a time to construct a derivation (Fig. 3). There is no real parser behind this tool, just string substitution, and the whole tool works in the client browser without any need to communicate with a server. It can surely be further developed to be combined with better visual display, and also evolve into a mechanism for step-running specific parsing algorithms.

Besides using special computer tools and understanding the algorithms, CL students often learn how to *program* language processing tools. The CL survey questionnaire did not ask *whether* programming was taught as part of a CL program of study, but *which* computer languages were employed. Prolog has 40 ad-

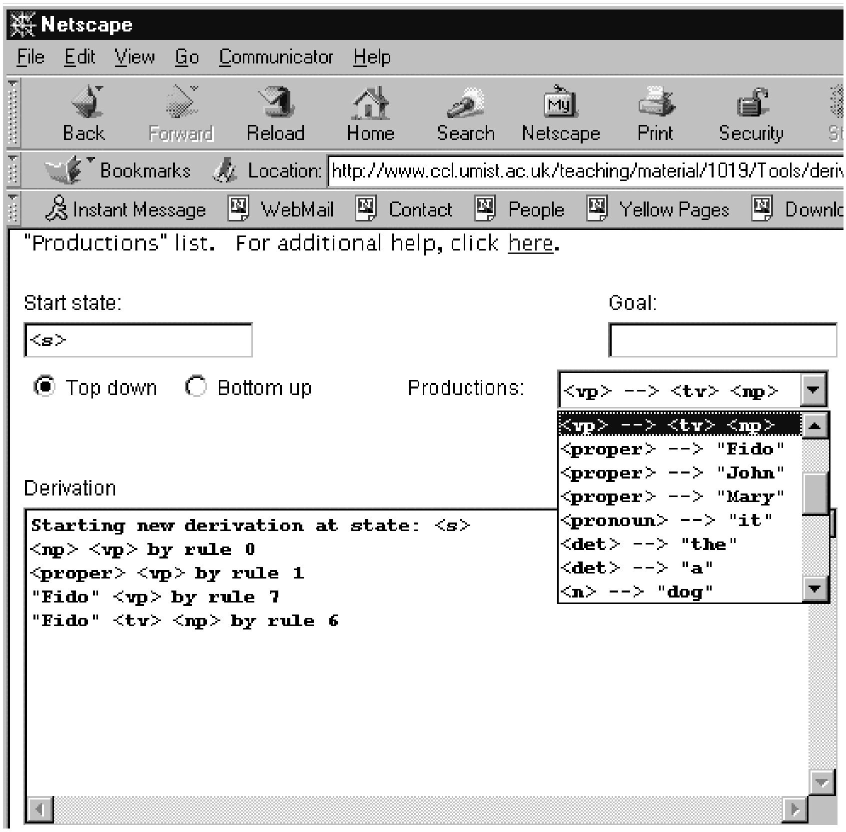

Fig. 3. A parser derivation tool, showing the steps in the analysis of a sentence, each step taken under user control.

herents among the respondents, more than twice as many as its nearest rival, but since that is Java, only a few years old, the picture may change dramatically in the near future. From the total number of responses, it is clear that programming languages provide the vehicle for much of the practical learning of computational linguistics techniques. The 102 responses show that on average 1.5 different programming languages are used per CL site. The use of 'ready-made' CL tools was nearly as widespread, but with the exception of WordNet, PC-KIMMO and the LFG-Workbench, no specific tool was used at more than 2 sites, although more than 50 distinct named and more unnamed packages are in use in teaching CL.

3.2 Philosophy and logic

As in computational linguistics, certain aspects of logic are extremely amenable to a computational treatment. The greater the degree of formalism that is achievable

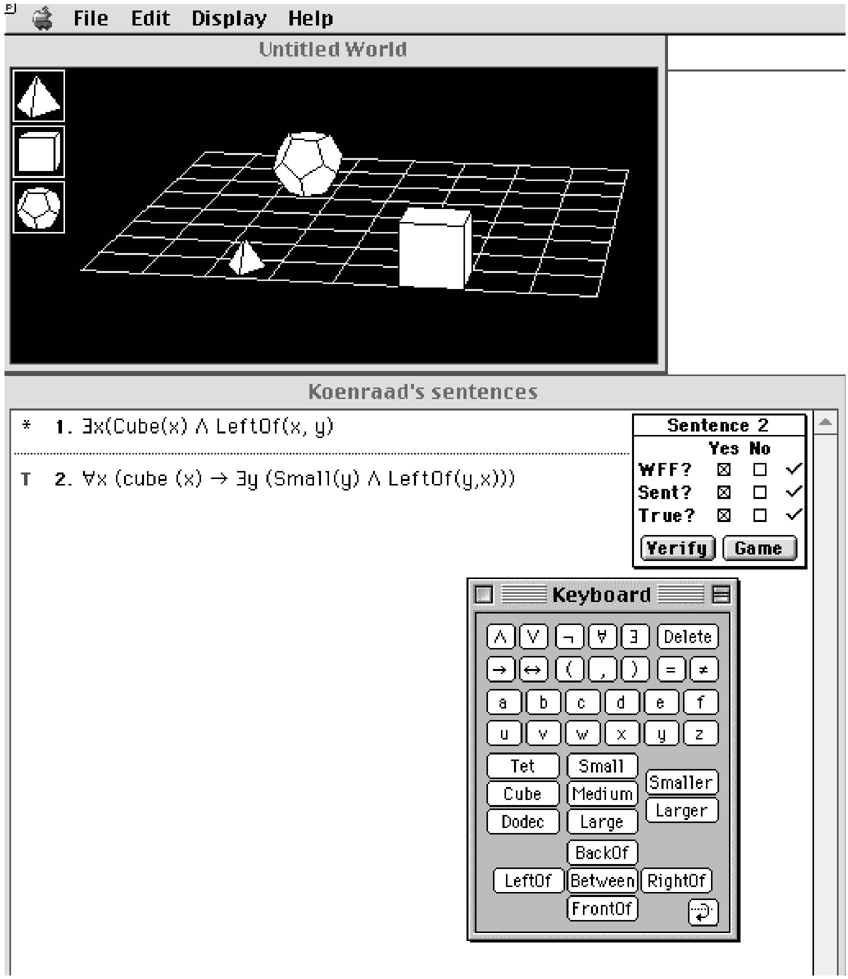

Fig. 4. Tarski's World screenshot.

in a field, the easier it is to generate some of the course material from a model of the subject expertise built into the software. That is, the subject knowledge should not be simply encoded as multimedia, but should be represented in the system on the knowledge level (Newell, 1982; Clancey, 1988). In scientific fields, this enables courseware to be developed as a *simulation* or a *model* of the domain in question. A theory of the domain is built into the program, and the student can explore the consequences of the theory to provide a more direct learning experience than that mediated by mere description no matter how well illustrated.

This has the benefit that once the machinery is in place, it is up to the student to provide the descriptive content and the teacher does not need to provide reams of text and pictures to generate content. In logic, one of the more successful computer-based learning packages is Barwise and Etchemendy's (1993) *Tarski's world* (`http://www-csli.stanford.edu/hp/Logic-software.html`, in which the student can develop descriptions and theories of a simple 3-D blocks world, which the student can view. The program helps teach not just the syntax of first order logic but also semantics, by determining whether the student's theory is consistent with the displayed world.

By means of exercises, students learn how to write well-formed sentences in this language and to determine the truth of such sentences. As Barwise & Etchemendy (1998) argue, it is advantageous for the student to be quickly identify and correct his or her own misconstrues of the language, rather than waiting for classroom help or, as happens all too commonly, getting though the entire course without the problem being noticed. To remedy this situation, Barwise and Etchemendy made their educational simulation program *Tarski's world*, for which they won the 1997 Educom medal.

Tarski's World helps students to acquire some mastery of the language of first-order logic. The program allows the student to construct and test logical sentences in relation to a world represented by simulated 3D objects on a screen (Fig. 4). Despite the simplicity of the program, the game-like nature of the simulation allows a flexibility which is hardly possible to achieve in a book. Students can evaluate given sentences with respect to any given world. Conversely, students can also construct a world that makes a set of sentences come true. Students can also make infinitely many new sentences to express facts about a world, and test these. The creative nature of the exercises is found enjoyable by students and stimulates the understanding of logical reasoning.

4 Courseware, multimedia, and hypermedia

Multimedia and hypermedia presentation and distribution of learning materials is increasingly popular, because the delivery media are now relatively cheap and efficient, and provide a convenient alternative for teachers to paper-based dissemination, as well as a slicker presentation mode. The principal current alternatives are high-density recoding media locally stored (CD-ROM) or distributed on the WWW. Both provide an essentially similar interface to the user, based on browsing interaction. The principal difference is that standard CD-ROM must be physically compiled and once written cannot be revised, whereas web pages are readily revisable by the author. Such differences have no intrinsic impact on the student experience. The degree of autonomy given to the student can vary to the extent that interactive devices are used by the courseware designer. Since both CD-ROM and the WWW provide ways to invoke programs, the degree of interactivity is restricted only by the courseware writer's imagination and by the convenience and usability of the authoring tools. The advent of cheap CD writers and media has been contemporaneous with the advent of the totally free WWW. Presently, CD-ROM is receiving less attention as a vehicle for courses than the Internet.

Multimedia can be used with any field of learning no matter how unstructured. The stimulus material can be any text, pictures, sound, video, animation, etc., and the role of the computer is primarily that of filing and presenting these resources. Such material takes a long time to prepare, and is usable only once (apart from repetition and reinforcement) per student. Consequently, the investment is considerable for fields with small student numbers. Nevertheless for those fields where the data is essentially visual or aural, the effort is often worthwhile.

Humanities disciplines are rapidly adopting presentation modes that exploit all the richness of multimedia and hypermedia in a pedagogical framework. Several such initiatives were started by MIT's Center for Educational Computing Initiatives (CECI, `http://www-ceci.mit.edu/`). The *Jewish Women's Archive* (`http://www.jwa.org/main.htm`), for instance, offers exhibits, a *virtual archive*, and educational resources presenting the rich legacy of Jewish Women from many perspectives, including historical, cultural, and social. JWA is not a plain collection of materials, but achieves its pedagogical aims by well-balanced multimedia presentations that are collages of text, portraits, facsimiles, transcriptions, audio recordings, and film clips. The site is browseable and searchable in several ways. Another CECI project, the *Virtual Screening Room* (`http://www-ceci.mit.edu/projects/vsr/`), is intended to provide a fully interactive multimedia environment for use in teaching *film analysis* at MIT and at other institutions. While conventional film analysis textbooks translate a kinetic medium into a static medium, the Virtual Screening Room allows the user to read about and view actual film clips in a single integrated environment.

Moving into the area of games, simulation, and interactive literature, *Operación Futuro* (`http://www-ceci.mit.edu/projects/of/`, also at MIT) offers interactive multimedia teaching to students of Spanish and South-American culture. In the module Historia Interactiva, the student participates in a non-linear interactive story where a young Colombian couple planning to get married has to find solutions to setbacks: money problems, work search, college costs, love and family relationships, etc. In order to help actively, students need to familiarize themselves to local conditions, culture, traditions, etc. and tests listening and writing skills. Somewhat similar, *CharacterMaker 4.1* (`http://www-ceci.mit.edu/projects/Eliza/`) is the first Java-based version of a program to create interactive characters. The software has been evolving as part of Janet H. Murray's course on Interactive Narrative at MIT (see also Murray, 1998). The aim of this program is to allow students to create interactive characters that carry on a conversation by scanning typed input for keywords, and replying with sentences associated with those keywords. In this context, we mention that also other humanities scholars are exploring ways in which the new technologies are causing 'literature' and 'film' to move towards non-stationary, non-linear, participatory, and multi-modal narratives. Moreover, the study of stories on digital media may be instrumental in a reflection on the reading process and on the non-static nature of even printed books. By way of example, we name Aarseth's (1997) explorations into *cybertext* and *ergodic literature*. It is felt that these developments will soon need to be incorporated into the curricula of *literary studies*.

In *speech communication sciences*, recorded and generated sound, as well as visual representations of analysed sounds, are the basic source materials for students to work with. The computer provides far more flexible ways of handling such materials than conventional recording devices like tape and video recorders. The SOCRATES thematic network project on Speech Communication Sciences has installed a group working on computer assisted learning (CAL) in this area, while also the International Speech Communication Association (ISCA, `http://www.isca-speech.org/`), has a special interest group on education. The group has conducted detailed studies on CAL packages available for the teaching of speech and CL (see Bowerman et al., 1999; Huckvale et al. 1997a, 1997b, 1998; Bloothooft et al., 1997-1999) and compiled a set of evaluation criteria which can be used as guidelines of best practice. The group organized a workshop on methods and tools for speech science education (see Hazan and Holland, 1999) and an *education arena* (`http://www.ling.umu.se/arena/`) for Eurospeech-99, which included a job fair and a presentation of courseware on CD-ROM and other media.

Clearly, scholars of music or *phonetics* can hardly be satisfied with textbooks. The use of computational techniques allows interactive and multimedia presentations, which are more useful than single-mode presentations. Consider, as a typical phonetics example, the *McGurk effect* (`http://www.media.uio.no/personer/arntm/McGurk_english.html`), which refers to the observation that when a person hears “ba”, while watching a face that says “ga”, the combined signal is interpreted as “da”. Obviously, the student can only fully appreciate this surprising effect when it is heard and seen, rather than when it is read from a book. Using an interactive video with a demonstration of the effect, students can experiment at will, including modes in which students either close their eyes or turn off the sound while observing the demo.

In phonetics, speech processing and communication disorders, the use of audio presentation and interactive sound processing has taken off. in Sweden the Centre for Speech Technology (at KTH Stockholm) has for instance developed *WaveSurfer* (`http://www.speech.kth.se/wavesurfer/`), a tool for recording, playing, editing, viewing, printing, and labeling audio data. WaveSurfer is suited for a wide range of tasks in speech research and education. Taking a different approach, Martin Cooke at Sheffield used *MATLAB* to develop a set of auditory and speech demos that can be downloaded (`http://www.dcs.shef.ac.uk/research/groups/spandh/MAD/docs/mad.htm`).

In another example, Klaus Fellbaum at the Brandenburg Technical University of Cottbus has developed a webcourse on Human Speech Production Based on a Linear Predictive Vocoder (`http://www.kt.tu-cottbus.de/speech-analysis/`), intended as a tutorial for the virtual university. The tutorial includes audiovisual demonstrations of the vocal tract function and, more impressively, it offers real-time speech analysis and resynthesis of the student’s own voice using advanced techniques. The main feature of the software, written in Java, is its numerous interactive functions that bring the demonstrations entirely under the student’s control.

Computational linguistics educators have also produced web mediated course materials covering a variety of aspects of their discipline. Respondents to the CL survey reported over 30 URLs to web courses. In addition, several other courses and tools have been announced at conferences and workshops (De Smedt & Apollon, 1998; Rosner, 1999). We found no instances of courses which simply use the WWW as a storage and distribution medium for textual notes. All use at least cross references via hyperlinks and other uses of the medium's interactive potential. Some courses include downloadable software or interactive demonstrations, often through CGI scripts, Java scripts, or Java applets that read input from a web page and deliver output in a browser window. Finally, although much attention is focused on the WWW, it must be noted that the Internet offers other modes of communication. The real-time online course by Dekker (1998) is a case in point for CL. Two concrete pilot courses n the web will be discussed, by way of example.

The pilot on *Statistical Natural Language Processing* (`http://www.ling.gu.se/˜nivre/kurser/wwwstat/`) is meant to provide the basic material for a distance learning course, although some local supervision or tutoring will normally be required. The content includes basic statistics, applied statistics (meaning Markov models and information theory), and natural language processing, covering language modeling, tagging and parsing, disambiguation, translation, and alignment. This content is partly based on previous work from Brigitte Krenn's and Christer Samuelsson's work *The Linguist's Guide to Statistics* (`http://fasting.hf.uib.no/papers/compendium.ps`), which in fact is the main text presented. In addition to this and other texts which are available online, the course presents a set of exercises with solutions for each topic on the course, a set of projects with all tools and data provided, slides for each topic, ad pointers to the literature. Finally, there are hyperlinks to practical tools and resources on the WWW. Because the course adopts a downloadable and printable book for most of the expository material, it does not fully exploit the web medium; the format of the presentation is kept simple. In the spring of 1998, the course was given as a distance learning course sponsored by *Computational Linguistics in Flanders* (CLIF, `http://clif.uia.ac.be/`), with Dr. Walter Daelemans as coordinator and Dr. Joakim Nivre as main lecturer. It must be noted that this course covers an urgent need for teaching materials in a new subfield of CL.

A course more clearly designed for an open and distance learning situation is the pilot on *Information Retrieval (mono- and multilingual) with Natural Language Processing techniques* (`http://rayuela.ieec.uned.es/ ircourse`) by Felisa Verdejo, Julio Gonzalo, and Anselmo Peñas at UNED. The course is usable standalone for self study, or as an adjunct to conventional courses. The course covers linguistic techniques in IR focusing on morphology, tagging, and multilingualism. The course uses the web medium well by allowing the student to actively use several computing tools via the WWW. In fact, the main pedagogical effort in the course is provide or assemble Internet online resources that permit not just reading, but also practical experimentation of the issues considered in the course. Such facilities include software at the site offering the course (stemmers, morphological analyzers, part of speech taggers for different languages, multilingual lexical databases, and cross-language mapping of queries) as well as external resources such as search engines, machine translation systems, etc. Experimentation is complemented with short introductions for every main topic, hyperlinks to reading materials available on the WWW, and self tests.

Even though tools such as these are valuable pedagogical instruments for basic concepts of CL, they may not address other important goals in present-day CL education. Bouma (1999) points out that many CL tools are aimed at the construction of *toy* systems, which are too far from reality. Instead, he proposes to offer a far more realistic learning context in which students are stimulated to make systems which account for actual linguistic data. This can be achieved by giving students easy access to large-scale resources such as extensive corpora and full-scale electronic dictionaries. Since students nowadays have access to hardware with sufficient computer power and data can be distributed over the net, there are no unsurmountable obstacles. Also, since it is possible to provide them with high-level tools, attention is not absorbed by low-level programming techniques. Instead, the student's minds are free to deal with the actual linguistic data. Eventually, Bouma hopes, this approach will enable students to deal with problems at a realistic level of complexity and prepare them better for actual challenges in human language technologies. Some of the concepts, projects, and tools used at Groningen can be seen online.

Large *language engineering* platforms are becoming more prevalent, such as *GATE* (`http://gate.ac.uk/`) at Sheffield, England (Cunninham et al 1997), in which complementary and alternative components can be integrated to produce modular text processing systems. Another powerful platform is the *CSLU* (`http://cslu.cse.ogi.edu/`) toolkit at the Oregon Graduate Institute (OGI) which is presently used in a course for designing and testing spoken dialogue systems (McTear, 1999). Using such platforms, students and developers do not need to develop basic tools for sub-tasks like parsing from scratch. Students will more and more be given a thorough grounding in how to use such platforms to develop comprehensive applications involving language and speech processing as part of more project-based education.

Web technologies - HTML document browsing combined with the opportunity to run programs at the server or client side - provide for arbitrary interactivity with the user, but often this requires many heterogeneous parts to be put together. However, computer-mediated learning with multimedia tends to use particular capabilities. At a basic level, there is the ability to manage and present the pedagogical materials, and this is already well catered for by standard web authoring tools, such as Netscape Composer, Macromedia Dreamweaver, and the most recent versions of most popular word processors. Multimedia courses make use of a number of additional facilities. For example, the ability to pose review questions or problems and compare student solutions against model answers; to track student progress; to run a discussion list, etc. All of these are possible using web technologies like CGI scripts, database connectivity, etc., but to use these raw tools is to undertake a large programming project. The alternative, to use a software environment for developing and delivering courseware is attractive to many educators, so much so that some of the software solutions can succeed in charging educational users per seat (e.g., *WebCT* (`http://www.webct.com/`), discussed in Goldberg, Salari & Swoboda, 1996). WebCT is quite indifferent to subject matter, with its main strength lying in student tracking and course management. It relies on standard web authoring tools for course content.

The ability to present linguistic analysis diagrams is an important feature of linguistic courseware - not to be underestimated as Bouma (1999) points out. Standard HTML lacks one important component, the ability to present diagrams generated dynamically, except as text.

The alternative is to run a program that can present line drawings within the browser. The Java programming language supports a range of graphical tools, the most primitive of which is the *canvas* on which lines can be drawn and mouse events reacted to. This is used in the tool by Black, Hill, and Kassaei (1999) shown in Fig. 5.

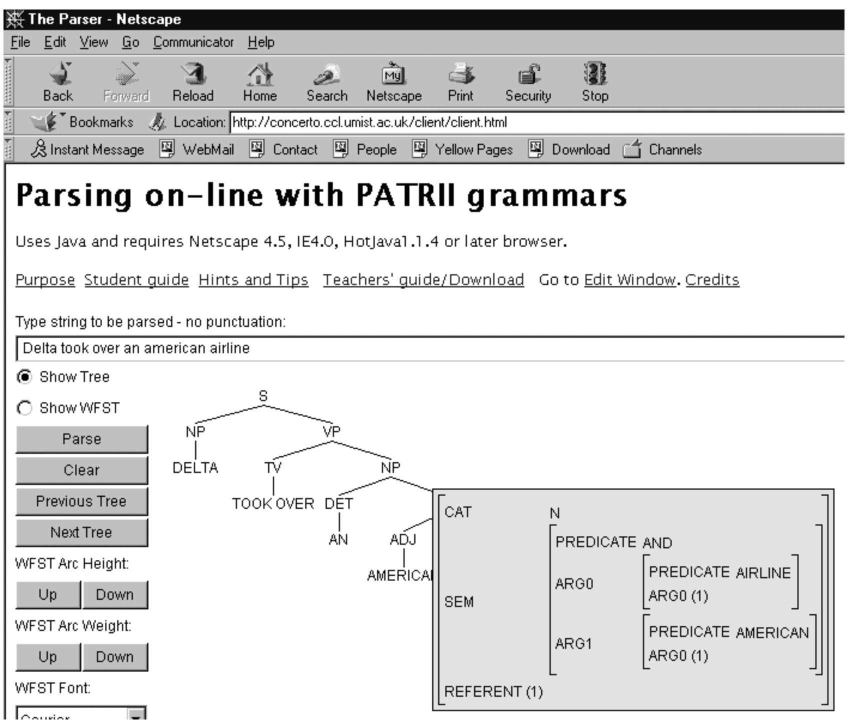

Fig. 5. WWW parser showing linguistic diagrams output in a Java applet's canvas.

An interesting tool by Calder (1998; 2000) that concentrates on the presentation and editing of linguistic diagrams is *Thistle* (`http://www.ltg.ed.ac.uk/software/thistle/index.html`). In Thistle, as well as viewing diagrams, the student can create and modify diagrams by direct manipulation of the nodes and labels in them (Fig. 6). Thistle diagrams are presented using consistent fonts and styles.

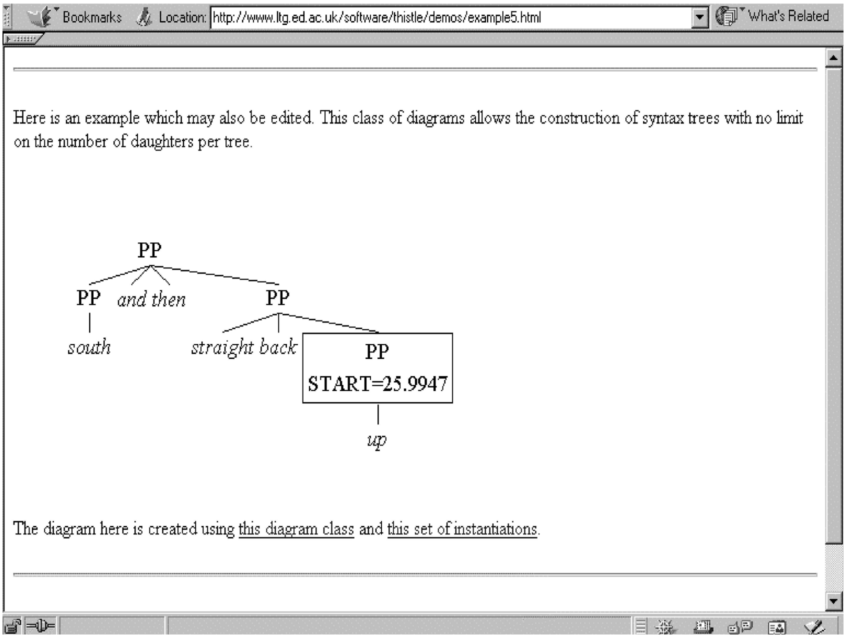

Fig. 6. Screen shot of the Thistle tree editor.

Of the demonstrable systems that CL instructors are using, the majority seem to be research-based systems that have been given a HTML-based front end, nearly always using server-side CGI scripts to conduct the processing. A popular alternative is to transfer the processing to the client, i. e., the browser. Scripting languages (like JavaScript) are favored by Gibbon and Carson-Berndsen (1999) because they are relative simple languages, and the student can inspect the source code. On the other hand, *applets* written in Java have a much more powerful and expressive language to draw on, greater standardization, graphics support, greater run-time efficiency due to compilation, and the ability to *hide* the source code from the student. Applets can run autonomously once downloaded, as with JavaScript.

However, the main reason that so many CGI based demonstrations are available is that server-side processing (in any language you like) makes it easier to re-use existing programs rather than writing them from scratch. Applets can access external resources, such as databases or legacy programs not written in Java to provide the user interface of a more complex and heterogeneous system. A system like this where processing is distributed between the applet and other programs residing on the server is referred to as a client-server system. The program by Black, Hill, and Kassaei (1999) depicted in Fig. 5 works in this way. The applet is downloaded from a web server embedded within an HTML file that contains supporting tutorial and user documentation. It opens a two way socket communication with a parser running on the same machine as the web server. The parser is written in Lisp, and was relatively little modified to work in co-operation with the user interface. Because an applet cannot load or save files on the local computer, an additional window is provided in which the student can edit grammars stored on the server.

Computer-aided learning (CAL) refers in general to the use of computers to facilitate learning. Several paradigms of CAL are in use, and observers usually place these on a continuum between those where the computer directs the student's learning and those where the student is enabled to undertake an open-ended exploration of a domain. Computer-directed programmed learning allows for individually paced student work through exercises and differentiation of stimulus and response material according to learning success. However, programmed learning materials are not dependent on recent developments in computer based multimedia, as this form of guided interaction can be built into programmed texts, adventure books, and even self-assessment tax returns. While CAL has enjoyed widespread use in adjacent fields such as language teaching and learning, it has been relatively less popular in CL prior to the advent of the WWW. The latter has stimulated much recent courseware development, such as that described above as web courseware. As in any field, there is much value in supporting the push of course materials with exercises and tests, which once developed can be repeatedly administered to each new cohort.

The learning goals we set for students in CL often include the development of theories of grammar for particular fragments of natural language. Programs that parse natural language strings according to a linguistic theory - a grammar and lexicon - written by the student also provide for the same kind of learning experience and immediate reinforcement in the domain of syntax. The Linguistic Instruments tools for the Macintosh by Linguistic Instruments, the LFG Workbench by Xerox, the programs by Bouma (1999) and by Black, Hill, and Kassaei (1999) referred to above can all be deployed to support the learning of linguistic formalisms in this way. Apart from the LFG workbench, all these recently mentioned systems are deliberately designed to be simple in use although they can support moderately realistic linguistic descriptions using unification based formalisms. Their authors have been particularly keen to support the needs of beginning students, and to dissociate the understanding of linguistic description from computer programming languages, so that the student can develop a cleaner mental model more readily.

We project that the current trends towards using the WWW and of information processing techniques will increase for all kinds of education, also for CL. As the demands for quality and pedagogical relevance increase, it will be recognized that many fields need special tools. But it is also important to give attention to the context in which CAL tools for CL are used. The ability to work in groups is important for CL education and especially since it is seeing its home in relation to other communities such as speech which involves much interdisciplinary interaction and expertise. Courses should stress the theoretical as well as the practical, project work, and the ability to work in groups.

5 Discussion and conclusion

Increasingly, the products of the human mind are available in digital form: digital libraries, online archives, museum web sites and other databases contain enormous amounts of material which is directly relevant to humanities scholarship. This increase, together with the increase in the power of personal computers, has created a situation where material in electronic form is longer reserved to a handful of researchers, but has come into reach of large numbers of humanities students. This is not to say that better availability makes the subject much easier. Language, art, and historical data are still difficult to interpret, and students have to become acquainted with ever more advanced techniques for analyzing them.

Although *humanities computing* is becoming established as a term, it covers an enormously diverse array of methods. Several factors contribute to this diversity: the large number of disciplines within the humanities, each with their own objects and methods, the different cultural settings in which institutions of higher education operate, and also the fact that the liberal arts are not bound by professional titles requiring tight regulations concerning accreditation, as also evidenced by the current lack of international coordinating forums for humanities education. The resulting wide diversity in the ways humanities students are introduced to information technology can be interpreted as a richness linked to cultural diversity, but it can also hinder students' chances for mobility. Humanities courses differ widely in their adoption of computing methodology. While some institutions offer basic computer literacy courses to their students, others confront their students more systematically and thoroughly with the methodological implications of computing in the humanities (De Smedt, Gardiner, Ore, Orlandi, Souillot & Vaughan, 1999).

Generally spoken, the multimedia products with the most clearly defined pedagogical strategy are those aimed at young children. Despite the tsunami effect of multimedia on society at large and up to now, very few dedicated learning products have been available for the training of students at higher education levels, and particularly in the humanities. This is unfortunate because advanced formal methods can hardly be taught from books alone, since their application requires not only access to electronic texts, but also advanced processing tools, preferably presented with appropriate pedagogical guidance. In general, industry is hardly interested in developing educational products for the humanities. An exception is the collecting, maintaining, and using textual data, which is one of the areas were academia and industry (e.g., academic repositories and commercial publishers) have demonstrated that there are ways to merge know-how and resources.

Creative, intelligent, exploratory learning tools have not yet reached widespread use in humanities education. In particular, the incorporation of such tools in humanities curricula has received insufficient support at all but department level. It seems that many current institutional, national and international strategies seem to prioritize ease of use over learning relevance and are hung up on keywords such as *multimedia* and *WWW*. Some limit the use of computers to the delivery of courses only, whereas others let humanities students use information technology actively and creatively. Clearly, the development and initial deployment of good exploratory learning tools needs investments as well as rethinking basic educational strategy. Specifically, the development of such tools relies heavily on advanced research in the humanities, cognitive science, and artificial intelligence.

The lack of available products is not surprising though, since humanities faculties are often dramatically under-equipped as far as new technologies are concerned (see, e.g., De Smedt, Gardiner, Ore, Orlandi, Souillot & Vaughan, 1999). This is of course a dissuasive element for publishing companies who might otherwise target this market sector. Also, for these companies and their multimedia designers, the subjects taught in the humanities (at university level) may appear too fuzzy or ill defined, involve too many related copyright issues, require too many experts or too much cooperation with university teachers. It follows that the evaluation of the return on humanities projects will generally suffer from a negative prognosis, and that the private sector will not invest much in the humanities under the present conditions.

Furthermore, there have been few teachers with enough motivation to acquire advanced training in new technologies. This seems less due to lack of interest than to the fact that keeping up with educational developments is not given as much academic credit as research. If the take up of multimedia by the humanities at university level is slower than in the rest of society, it is not because of any reluctance on the part of teachers. Nevertheless, many experiments in web teaching are being conducted in the humanities (e.g., the Oxford-based Virtual Seminars project at `http://info.ox.ac.uk/jtap`).

As a case in point, a report on the teaching of Japanese in Europe revealed that when the subject is taught at humanities faculties, the integrated use of new technologies in learning this language is far lower than when the same subject is taught at business or technology schools. Students of Japanese at humanities faculties generally have limited access to computers, or no access at all. When some computers are available, the time students can spend on them is much too short, which is deeply frustrating, since many students have limited skills as regards basic computer tools.

Students need to acquire basic computing proficiency. However, this should be gained before or outside their humanities studies. This raises the question of how institutions should provide it, at least until they can expect that every student will arrive from secondary school with the necessary skills. There are a number of models for the inclusion of more methodological computing components in textual studies curricula, with two important types of model, broadly speaking. In the first of these, the interdisciplinary aspects of the appropriate formal methods and techniques form the basis of components that are taught to students in a number of disciplines; in the second, those computing techniques and tools that are held to be important for a specific discipline are included in courses offered in that discipline. Both approaches have their particular advantages, and in many cases they are related to institutional history and culture. However, in both cases some important questions that need to be addressed before meaningful choices can be made:

- Who should teach the computing components?

- To what extent should the students be taught the *application* of the appropriate techniques, and to what depth should they be expected to understand the underlying computing or other technical *principles* involved?
- To what extent should the acquisition of *transferable skills* determine the scope and scale of such courses and programs?

Furthermore, the Internet explosion offers particular opportunities for students who are used to deal with multimedia resources, to mix text and images (and sounds), and who are proficient to create and manipulate such resources. This is an area of activity that raises the *transferable skills* issue in a particularly acute form. A key aspect of computing in the humanities is the need to develop students' analytical skills, which in turn make them particularly attractive in the wider labor market. This raises issues of how to ensure that prospective employers understand that the students have more to offer than merely the ability to use, e.g., a database or a spreadsheet package.

One of the first and most obvious applications of computing in textual disciplines was the creation of concordances. This technique remains important, of course, but is relatively straightforward, and different types of concordance can be generated by a number of readily available software tools. Still, it is remarkable that many students of literature and textual sciences today are still not being introduced to actively using such tools. Learning to use such new computational tools is simply not a part of the curriculum. Nevertheless, the availability of user-friendly statistic packages makes it possible to consider the inclusion of statistical methods in all humanities education, including the textual disciplines. One question raised by this idea is whether the general benefit to society of having truly numerate citizens is outweighed by the difficulties of applying measurements to humanities data in an appropriate way and the possible resistance to the idea on the part of humanities students.

The technological opportunities afforded for multimedia in education are only one of the factors that have stimulated the recent development of prototypes for computer mediated education in the humanities. Many countries have open universities which specialize in distance learning across a range of disciplines, and these institutions have designed their structures and procedures around the needs of part time correspondence students. A principal difference between these institutions and conventional universities is that distance learning is impossible without a high investment in the development of publishable quality teaching materials together with systems for student feedback and assessment.

Distance learning is also seen in conventional universities as a significant proportion of their academic activities in the 21st century. For specialist areas in the humanities, distance learning appears to offer a solution to the numbers problem. It is rare to find more than a half a dozen specialists on computational linguistics or medieval manuscripts in one university, and therefore the staff to cover a broad curriculum in the discipline. Hence inter-institution collaboration in the provision of distance learning modules is a possible way to make study units available where the local expertise is lacking.

The course on statistical natural language processing by Joakim Nivre (http://www.ling.gu.se/ nivre/kurser/wwwstat/index.html) is a pertinent example. Looking at the programs of recent conferences, it is clear that the total of only four courses on statistical methods listed in answer to the CL questionnaire reflects a mismatch between what is currently offered to students and the CL research agenda. Nivre's course may well answer to a widespread need that is going unanswered not because it is unrecognized but because of the start-up costs in initiating new modules outside of the teacher's core specialism. On top of that, the problem of funding for maintenance and making things robust is even more difficult than for exploratory work.

Computer-mediated learning in general, like more traditional correspondence courses, removes the temporal and spatial requirements on study, by making learning material available to students irrespective of time and place. Nevertheless, most distance learning institutions and experiments find it helpful to place temporal if not spatial restrictions on the students. If students are to be accredited by distance learning institutions, there is more need than in conventional universities for students to be examined in invigilated time-constrained examination conditions, because the opportunities for plagiarism and cheating are obvious.

There is also a social dimension to learning, in which solidarity with fellow students is an important motivating and sustaining factor. Distance learning institutions often try to encourage this by both formal tutorials and informal self-help arrangements. One particularly interesting experiment in distance learning in our discipline was conducted by Dekker (1998) who arranged a virtual classroom using an email list server at set times each week for a group of students of logic around the globe. These experiences suggest that computer mediated learning in general need to be seen as a socio-technical system, for use in a particular institutional context.

Summing up, the current status of computing in humanities education is advancing and has good potential for further development. Nevertheless, initiatives so far have been isolated and have insufficient leverage. In order to gain momentum and prevent reinvention of wheels, co-operation between institutions is necessary. An example of cooperation has been set by the SOCRATES thematic network project on Advanced Computing in the Humanities, which organized a major conference and produced surveys and publications on the theme. Other forums have also been established, for instance as education sessions tagged on to international research conferences, including the ACL-ALLC, CHArt and EACL conferences. Nevertheless, intensified and more long-lasting networking would be welcome.

Bibliography

Aarseth, E. (1997). *Cybertext: Perspectives on Ergodic Literature* http://www.hf.uib.no/cybertext/. Baltimore: Johns Hopkins University Press.

ACO*HUM CL Survey (1999) *Advanced Computing in the Humanities (ACO*HUM)* http://www.uib.no/acohum

Barwise, J. & Etchemendy, J. (1993). *The Language of First-order Logic (3rd Edition).* Stanford: CSLI Publications.

Barwise, J. & Etchemendy, J. (1998). Computers, Visualization, and the Nature of Reasoning. In T. W. Bynum and James H. Moor (eds.), *The Digital Phoenix: How Computers are Changing Philosophy* (pp. 93-116). London: Blackwell.

Bergen University Department of English (2000) *The Bergen Corpus of London Teenage Language (COLT)* http://www.hit.uib.no/colt/ University of Bergen, Norway.

Black, W. J., Hill, S. & Kassaei, M. (1999) Intranet learning tools for NLP. In Rosner, M. (Ed.), *Computer and Internet Supported Education in Language and Speech Technology. Proceedings of the Workshop, EACL '99*, Bergen, 12 June 1999.

Bloothooft, G. et al. (Eds.) (1997-1999) *The landscape of future education in speech communication sciences*, Vol. 1-3. Utrecht: OTS.

Bouma, G. (1999). A Modern Computational Linguistics Course Using Dutch. In Rosner, M. (Ed.), *Computer and Internet Supported Education in Language and Speech Technology. Proceedings of the Workshop, European Conference on Computational Linguistics, Bergen: EACL '99*, 12 June 1999.

Bowerman, C. (1999). Criteria for evaluating internet tutorials in speech communication education. In *Proceedings of the Sixth European Conference on Speech Communication and Technology (Eurospeech-99)*, Budapest, Hungary, September.

Calder, J. (1998). How to build a (quite general) linguistic diagram editor. In *Thinking with Diagrams (TwD98)*, Aberystwyth, UK.

Calder, J. (2000). Interarbora and Thistle: Delivering Linguistic Structure by the Internet. In *Proceedings of the Second International Conference on Language Resources and Evaluation. Volume II*, Athens, Greece, 31 May - 2 June, 2000, 1163–1166.

Clancey, W. J. (1988). The role of qualitative models in instruction. In Self (Ed.), *Artificial Intelligence and Human Learning: Intelligent Computer-Aided Instruction* (pp. 49-68). London: Chapman and Hall.

Cooke, M. (1998) *MATLAB Auditory & Speech Demos* Speech and Hearing Group, Department of Computer Science at the University of Sheffield. http://www.dcs.shef.ac.uk/research/groups/spandh/MAD/docs/mad.htm

H. Cunningham and K. Humphreys and R. Gaizauskas and Y. Wilks. Software Infrastructure for Natural Language Processing. In *Proceedings of the Fifth Conference on Applied Natural Language Processing (ANLP-97)*, Mar, 1997. http://xxx.lanl.gov/abs/cs.CL/9702005

Dekker, P. (1998). Teaching 'live' on the internet. In De Smedt, K. & Apollon, D. (Eds.), *The Future of the Humanities in the Digital Age: Problems and perspectives for humanities education and research: Abstracts. University of Bergen, Norway, September 25-28, 1998* (pp. 66-69) http://www.uib.no/acohum

De Smedt, K. & Apollon, D. (Eds.) (1998). *The Future of the Humanities in the Digital Age: Problems and perspectives for humanities education and research: Abstracts*. University of Bergen, Norway, September 25-28, 1998. http://www.uib.no/acohum.

De Smedt, K., Gardiner, H., Ore, E., Orlandi, T., Souillot, J. & Vaughan, W. (eds.) (1999). *Computing in humanities education: A European perspective*. Bergen: University of Bergen, HIT centre. http://www.hit.uib.no/AcoHum/book/

Fellbaum, C. (ed.) (1998). *WordNet: An Electronic Lexical Database.* Cambridge: MIT Press. http://mitpress.mit.edu/book-home.tcl?isbn=026206197X

Fellbaum, K., Richter, J. (1999) *Human Speech Production Based on a Linear Predictive Vocoder*. Brandenburg Technical University of Cottbus. http://www.kt.tu-cottbus.de/speech-analysis/

Gibbon, D. & Carson-Berndsen, J. (1999). Web tools for introductory computational linguistics. In Rosner, Michael (Ed.) *Computer and Internet Supported Education in Language and Speech Technology. Proceedings of the Workshop, European Conference on Computational Linguistics, Bergen: ACL*, 12 June 1999.

Goldberg, M. W., Salari, S. & Swoboda, P. (1996). World Wide Web Course Tool: An Environment for Building WWW-Based Courses, *Computer Networks and ISDN Systems, 28.*

Huckvale, Mark, C. Benoit, C. Bowerman, A. Eriksson, M.Rosner, M. Tatham & B. Williams (1997a). Opportunities for computer-aided instruction in phonetics and speech communication provided by the internet, In G. Kokkinakis, N. Fakotakis, and E. Dermatas (Eds.), *Proceedings of the Fifth European Conference on Speech Communication and Technology (Eurospeech-97), Rhodes, Greece, September, Vol. 4*, 1919-1922. European Speech Communications Association (ESCA), Grenoble, France and University of Patras, Greece.

Huckvale, Mark, Benoit, Christian, Bowerman, Chris, Eriksson, Anders, Rosner, Mike, Tatham, Mark & Williams, Briony (1997b). Computer-Aided Learning and use of the Internet. In Bloothooft et al. (eds.) *The Landscape of Future*

Education in Speech Communication Sciences, Vol 1: Analysis. Utrecht, The Netherlands: OTS Publications.

Huckvale, Mark, C. Bowerman, A. Eriksson, M.Rosner & M. Wolters (1998) Catalysing web-based teaching with tutorial components. In De Smedt, Koenraad & Daniel Apollon (Eds.), *The Future of the Humanities in the Digital Age: Problems and perspectives for humanities education and research: Abstracts. University of Bergen, Norway, September 25-28, 1998* (pp. 70-72). `http://www.uib.no/acohum`

Krenn, B. & Samuelsson, C. (1997) The Linguist's Guide to Statistics. `http://www.coli.uni-sb.de/~krenn/stat_nlp.ps`

M. McTear. (1999) Using the CSLU Toolkit for Practicals in Spoken Dialogue Technology. In *Proceedings of ESCA/SOCRATES Workshop on Method and Tool Innovations for Speech Science Education*, London, UK, Apr 1999.

Murray, J. (1998). *Hamlet on the Holodeck.* Cambridge: MIT Press.

Murray, J. H. and Privat, M. (1998) *CharacterMaker 4.1*, Center for Educational Computing Initiatives, Massachusetts Institute of Technology. `http://www-ceci.mit.edu/projects/Eliza/`,

Newell, A. (1982). The knowledge level. *Artificial Intelligence 18,* 87-127.

Oxford University Computing Service (1999) *British National Corpus (BNC)* `http://info.ox.ac.uk/bnc/`, Computing Services, Oxford University, England.

Rosner, M. (Ed.) (1999). *Computer and Internet Supported Education in Language and Speech Technology. Proceedings of the Workshop, EACL '99*, Bergen, 12 June 1999.

Verdejo, F., Gonzalo, J. and Peñas, A. (1999) *Information Retrieval and Natural Language Processing.* `http://rayuela.ieec.uned.es/~ircourse/`

Index