Why should linguists compute?

Reflections on language documentation and linguistic theory

Dafydd Gibbon

Universität Bielefeld

Version 2007-11-18

Table of Contents

Index of Tables

Index of Figures

1 Why focus on computers in linguistics?

Perhaps it is not obvious to many why one needs – within certain strictly set conditions – to focus on computers and computing in linguistics, including neighbouring disciplines from phonetics through psycholinguistics to sociolinguistics.

An objection often raised is: computers do not contribute anything which human beings cannot contribute, except speed and quantities of data. There are a number of counter-objections to this.

First, a computer is not a *deus ex machina*, a being from on high with supernatural powers to influence the natural course of things. A computer is a product of the human intellect; if a computer contributes anything to science, this means that there are human intellects behind the computer to

which this contribution is due. A lot of anthropomorphic imagery is used in connection with computers, particularly when "they do not behave", and cause frustration. But the fact remains that the computer is a tool, an artefact constructed by human beings – a complex one, but still an artefact. This is what computers as we know them will always be.

Second, there are situations where more speed and larger quantities lead to a qualitative leap. The simplest case is in sports or making music: if a player does not have control over his speed, he will fail, a qualitative distinction. Catching a bus, train or plane is another case in point – if you miss it, you miss it, even if it is only by a second or two, and even if you ran very fast; this is a qualitative difference. Similarly, in linguistics, a large quantity of data will stimulate more and better insights than a small quantity, and the use of computers to analyse collections of data provides the foundations for yielding these insights and systematising them, a qualitative difference.

Third, computing enforces criteria of *consistency*, *completeness*, i.e. avoidance of undergeneralisation, rather than selective picking of isolated phenomena, and *soundness*, i.e. avoidance of overgeneralisation.

An analogy with familiar physical tools such as spectacles, magnifying glasses and telescopes may make things clearer: without these devices to make more detail available for human inspection, many qualitative insights about the world would be missed. Again, ropes and levers, not to mention machines driven by fossil or nuclear energy, or by natural forces such as water and air, are fundamentally quantitative extensions of human powers, but make qualitative progress possible.

2 Linguistic computing

In linguistics, the leap from quantitative processing to qualitative improvement has been particularly noticeable in a number of technologies. The following is a loosely ordered list of some of these areas:

1. Phonetics – the use of signal analysis software, together with statistical software, for achieving insights into the structures of speech.
2. Phonetics – the use of signal analysis and visualisation software (such as Praat, WaveSurfer, Transcriber and many others) for examining details of individual spoken utterances.
3. Descriptive linguistics – the use of text corpus analysis in corpus linguistics, both to verify hypotheses about the use of language in many contexts, or to induce patterns by means of statistical methods as a basis for further theory formation.
4. Descriptive linguistics – the use of parser software based on linguistic theories, in order to confirm or refute predictions about syntax and morphology.
5. Descriptive linguistics – the use of logical programming techniques, in order to model the understanding of language and to study the way in which we make semantic and pragmatic inferences.
6. Natural Language Processing – an extension of compurational corpus linguistics and linguistic modelling to the computational analysis of texts (text mining) and generation of texts, and in corpus-based computational lexicography.
7. Speech Technology – an independent engineering discipline which derived from Physics, and in particular from Acoustic Engineering, which has applied linguistic concepts for modelling sounds (phonemes, phoneme combinations called diphones, triphones, etc., syllables), pronunciation patterns in the lexicon, and morphological and syntactic language models in both Automatic Speech Recognition and in Text-To-Speech synthesis software.

In the following sections, a number of cases will be picked out, some perhaps unexpected, in order to illustrate the point.

3 Linguistic foundations of the internet

3.1 The internet as text

The internet consists of huge numbers of texts linked together and linked to multimedia objects. All of these texts must be consistently formatted in order for computers to process them. These consistent formats are based on text grammars – *Document Type Descriptions,* DTDs – originally developed by linguists, librarians, company documentarists and archivists in the 1980s. In fact, the metalanguage in which these grammars are described, SGML, the *Standard Generalised Markup Language*, has had the status of an international standard: ISO 8879:1986. A special case of SGML is HTML, *Hypertext Markup Language*, which is used for describing documents on the World Wide Web. There are many special cases as there are document types; these have been being systematised since the late 1980s by the *Text Encoding Initiative*, a group of linguists and computational linguists, since then, in order to permit documents to be digitised and archived systematically.

3.2 Text grammars and the internet

Of course the average library or internet user does not notice, or know, or want to know about these things. But these intellectual and physical tools are essential parts of our science, and, in turn, have their own foundational linguistic and scientific features. For example, it is little known, and deserves wider knowledge, that these web documents use the Phrase Structure Grammars invented by Chomsky in the 1950s, and that the HTML descriptions consist of labelled bracketings which are entirely analogous to the labelled bracketings used in linguistics. The notation is a little unfamiliar, but this makes no difference to the facts. A simple example will illustrate this.

Figure 1 shows a simple text, formatted in HTML, as shown by the Firefox internet browser. The top line of the figure is the title bar, which is part of the *metadata* of the document; this information, like library catalogue data, is not part of the actual document body, and is defined in the head of the document – what librarians call the "front matter". The next four lines show typical browser menu functions, and then the main field of the window shows the actual text. The bottom line is again a typical browser information line.

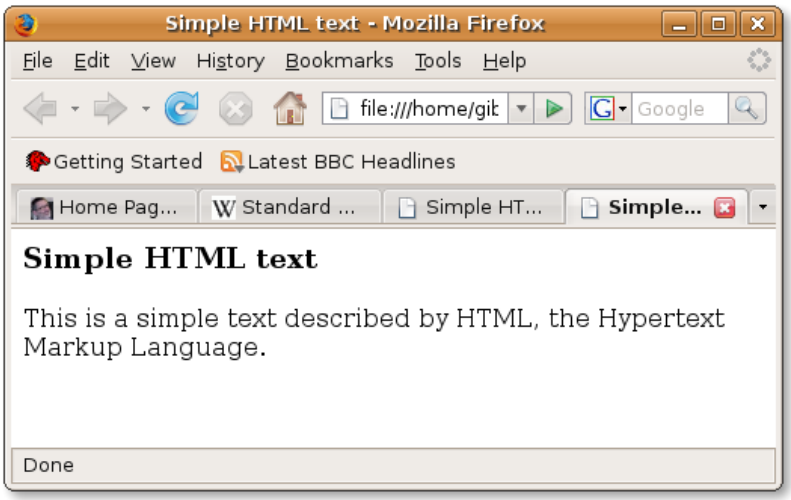

Figure 1: Simple text formatted in HTML.

Table 1 shows the HTML Structural Description (SD) of this text, with the embedded constituents indented in order to visualise the structure – "prettyprinted", to use the jargon term. The prettyprinted format clearly shows the tree-structure embedding of the parts of the document;

this is exactly analogous to the hierarchichal structuring of sentences or words, though the constraints on text grammar, and the categories involved, are clearly not the same. The structure can be represented equally well by means of a tree diagramme or a labelled bracketing.

In the table, the top category is "html"; the main constituent categories are "head", for a description of metadata such as the title, and the "body" for a description of the text. Within the body, there are just two units: a heading "h3" of a particular kind, and a paragraph "p".

Table 1: Structural Description (SD, labelled bracketing) of the simple text, described by the HTML text grammar (Document Type Description, DTD).

```
<HTML>
        <HEAD>
                <TITLE>
                Simple HTML text
                </TITLE>
        </HEAD>
        <BODY>
                <H3>
                Simple HTML text
                </H3>
                <P>
                  This is a simple text described by HTML, the
                  Hypertext Markup La nguage.
                </P>
        </BODY>
</HTML>
```

Table 2 shows the Phrase Structure Grammar (Document Type Description) from which the SD shown in Table 1 is derived. The Phrase Structure Grammar is, formally, of exactly the same kind as the Chomskyan Phrase Structure Grammars used in linguistics. This is the grammar of HTML (more exactly: of a small part of HTML, because documents are in general much more complex). The full HTML grammar describes any document which can be found on the World Wide Web, and is used by browser software like Firefox or Internet Explorer in order to parse the HTML description and to convert the resulting parse into a conventional print-like document rendering.

Table 2: Text linguistic Phrase Structure Grammar for the HTML Structural Description of the simple text, with Structural Description in a common linguistic notation.

```
HTML        →  HEAD BODY

HEAD        →  TITLE

TITLE       →  Simple HTML text

BODY        →  HEADING PARAGRAPH*

HEADING     →  Simple HTML text

PARAGRAPH   →  This is a simple text described by HTML, the
               Hypertext Markup Language

(HTML (HEAD (TITLE Simple HTML text) )(BODY (HEADING Simple HTML text)
(PARAGRAPH This is a simple text described by HTML, the Hypertext
Markup Language.)))
```

The prettyprinted version is standard practice in the documentation community, and is much easier to read than the standard linguistic notation. However, the two are totally equivalent in the sense that they can be uniquely translated into each other. The example provides a clear illustration of a Text Linguistic modelling process. If formal and computational linguistic methods had not been around, the internet would not have been born.

4 Language description and language documentation

4.1 Documentary Linguistics

So how do all these considerations relate to language documentation? Essentially, it is the "bread and butter" of language documentation.

For about fifteen years, a new paradigm in descriptive and applied linguistics has been gradually establishing itself: Documentary Linguistics. The striking feature of Documentary Linguistics is the value it attaches to *attested* and *authentic* linguistic evidence ("AA evidence"), and its aim to provide *comprehensive*, *effective*, *state-of-the-art*, *affordable* and *fair* documentary record of this evidential basis ("CESAF" criteria). The roots of Documentary linguistics lie in three main areas:

1. Field linguistics and the study of endangered languages, with the need to document and archive text (including recordings and transcriptions), dictionary and grammar resources.
2. The text modelling needed for classification and archiving of printed matter and software, as discussed in connection with the internet, in consistent descriptive text linguistic formats.
3. The quantitative empirical methods of speech engineering, with large quantities of text and speech data needed for inducing generalised models of language and speech.

In effect, Documentary Linguistics attempts to provide Descriptive Linguistics with the quantity and quality of data which other sciences are accustomed to, particularly the natural sciences. Lexicography has been the main linguistic science which has utilised large quantities of data, from the beginnings of modern lexicography in the mid-19th century to the corpus-based COBUILD series of Collins dictionaries in the late 20th century, and now practically all serious dictionaries.

However, many areas of Descriptive Linguistics have either chosen or been forced to restrict their attention to rather tiny selections of data. This is particularly characteristic of some of the more formal structuralist frameworks, in which evidential quality was reduced to simple intuitions of *acceptability* and *similarity*, and data sets were reduced to small sets of more or less systematically related sentences. The results from these methods have been impressive, but they have their limits: can the results from the analysis of small sets of citation form uses be transferred to authentic uses of language outside the laboratory or the study?

Similar considerations apply to the traditional experimental varieties of linguistics, including psycholinguistics, where authenticity of data has to be sacrificed to systematicity, with consequent problems of "ecological validity" - can the experimental results be transferred to authentic uses of language outside the laboratory?

New perspectives of precision and efficiency have been introduced into the documentation of language and speech from the field of speech technology, which shares with linguistics the need to develop effective phonetic, morphological, syntactic, lexicographic, textual and discoursal models. Extensive research has been done over the past 40 years on the economically well-supported languages; research on African languages is increasing, most prominently in South Africa by Justus Roux and his team, but also in East Africa (e.g. Amsalu & Gibbon 2005) and in West Africa (e.g. Gibbon, Urua & Ekpenyong 2004; Gibbon & Urua 2006; Gibbon, Urua & Ekpenyong 2006).

4.2 Objects and methods of Documentary Linguistics: the WELD approach

In cooperation with colleagues at the Université de Cocody, Abidjan, Côte d'Ivoire, and colleagues at the University of Uyo, since around 1997 the Bielefeld team has extended previous cooperation in the area of fieldwork to cooperation in Documentary Linguistics projects, funded by the Deutscher Akademischer Austauschdienst, the Deutsche Forschungsgemeinschaft and by the Volkswagenstiftung. In the course of these projects, the *Workable Efficient Language Documentation* specialisation of the overall language documentation paradigm was developed (Gibbon 2003). A feature of this paradigm is that easily manageable computational techniques are used in order to make language documentation more workable, and to provide a larger, more coherent basis for descriptive linguistic work.

The objects of language documentation procedures are, in general, the three traditional types recognised as basic in linguistic fieldwork:

1. Texts: collections of texts of any kind, traditionally narrations of stories, but also including transcriptions of speech recordings, from individual pronunciation examples to task-oriented dialogues.
2. Dictionaries: generally collated these days as a database (though paper notebook form may be used for field notes in preparation for making a dictionary) in the form of
    1. a simple table in a *word processor* such as OpenOffice Writer or MS Word (though these are liable to many inconsistencies which make computational processing difficult);
    2. a more comprehensive table in a *spreadsheet*, such as OpenOffice Calc or MS Excel (which is a great step forward in comparison to a word processor table and therefore to be recommended over these);
    3. a properly structured *database management system*, with supporting facilities for including more varied lexical information, and for automatically producing nicely formatted printed dictionaries.
3. Sketch grammars: ranging from phoneme tables and morpheme lists to systematic functional and/or structural grammars, either informally though systematically described, or formally precise.

The techiques used in language documentation of these kinds of objects range from speech signal recording and the annotation of audio and video recordings of these recordings, through computational support for interview prompting and lexicon construction.

In the Uyo projects, using such methods a number of interesting and widely acknowledged results have been achieved, in particular:

1. A collection of consistently transcribed texts of Ibibio (note: most everyday texts on computers are not consistently typed – they often consist of fanciful combinations of fonts which are used inconsistently and are therefore almost impossible to convert to modern formats).
2. A large machine readable dictionary of Ibibio which has been taken up by an Ibibio expatriate in the USA who is a software engineer and has converted the dictionary into a web-based hypertext dictionary or hyperlexicon.
3. A large concordance derived from the texts as a basis for linguistic work.
4. A prototype Text-To-Speech (TTS) synthesiser for Ibibio.

5 Documentary and descriptive linguistics in context

Documentary linguistics is not a field on its own, or just an arbitrary intermingling of other fields in an interdisciplinary context – though it is highly interdisciplinary – but it is intimately interwoven with and justified by relationships with other areas of linguistics. The main relationships are illustrated in Figure 2.

The model shown in Figure 2 is straightforward, and many details can be added. But the structure of the model is familiar: the two columns represent the distinction between Theoretical Linguistics on the one hand, and Applied Linguistics on the other. The three rows represent levels of abstraction in science, which were perhaps most succinctly formulated by Chomsky in the 1960s as *observational adequacy* (pertaining to the bottom row), *descriptive adequacy* (pertaining to the middle row) and *explanatory adequacy* (pertaining to the top row), in other words, representations of the *data*, representations of linguistically significant *generalisations*, and representations of evaluation procedures for the *comparison* of descriptions. The double-headed arrows indicate that there is not a one-way track from theory to application, but that theoretical development is also stimulated by the problem-solving processes involved in creating applications.

The parallels in the theory and application oriented columns demonstrate that applications are as sophisticated as theories – development procedures for lexicons or speech synthesis systems, for example, involve the same levels of abstraction. In speech engineering there are competitions run

by funding agencies to determine which system, at the middle level of the model, is best: for speech synthesis systems the criteria which are applied in statistically evaluated experiments are *comprehensibility* (the most important criterion) *naturalness* (whether the system sounds like real speech, and *acceptability* (whether the system is fit for purpose).

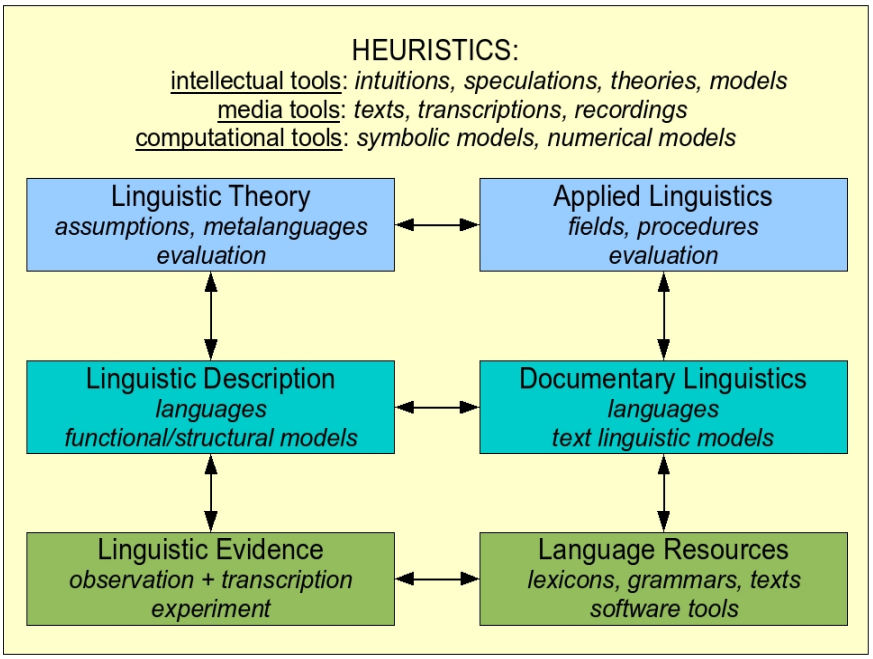

Figure 2: Language description and language documentation in the context of Theoretical and Applied Linguistics and their empirical foundations.

6 Kinds of linguistic computing

The preceding sections have been rather abstract. What added value does the linguist get from linguistic computing? First, it will be useful to distinguish different kinds of linguistic computing, and then to go into a selection of case studies which are relevant for theoretical and applied work on African languages.

6.1 Types of "computer"

Until the 1930s, the term "computer" was agentive, and meant "one who computes"; the instrumental meaning came in the 1940s with the development of the digital computer. The main distinctions can be characterised as follows:

1. *The general computer user*:
    1. Internet. Pretty much everyone uses the internet for information exchange and coordination of cooperation. The various internet media could however, be exploited much more fruitfully:
        1. Emails are very common, but for maximum effectivity require easy and reliable access, which is not given everywhere.
        2. Where internet facilities are reasonably readily available, "chat", i.e. written dialogue via the internet, is a very useful way for rapidly discussing both scientific and organisational issues.
        3. There are chat facilities, such as Skype, which offer additional services such as flexible exchange of files during a chat dialogue, as well as a "Voice over Internet

Protocol" (VoIP) telephone-like service (which is free between two computers, and relatively inexpensive between computers and telephones), and a video mode for the telephone services (which requires a broadband connection).

2. Office software. The professional linguistic computer user typically uses standard office software:
    1. word processor for writing articles, teaching materials, and administrative documents (note that linguists are heavily represented on the payrolls of Microsoft, Google and many other large companies, for developing text format styles, spell checkers, grammars, dictionaries, translation software, and for software localisation);
    2. spreadsheet for financial calculations and simple databases.

2. *The specialised computer user*:
    1. In phonetics, specialised software is available for many purposes, both for *illustration* of specific points, and for experiment and statistical *analysis* of speech.
    2. In lexicography, the user of specialised lexicographic data acquisition software; in linguistics, perhaps the most widely used software is SIL's *Shoebox*, or the later development *Toolbox*.
    3. Other software, such as morphological analysers, taggers which supply linguistic descriptions to texts for the purposes of Documentary Linguistics.
3. *The computational linguist as programmer or software developer*:
    1. The corpus linguist, who uses "scripting languages" for rapid prototyping of techniques for extracting tokens of linguistic units from texts and transcriptions, and analysing their distributions in order to determine phonological, morphological, syntactic and textual patterns. This may be done either for theoretical or for practical purposes.
    2. The model-builder, who uses Artificial Intelligence techniques to build parsers and generators, either for testing theories against data, or for building natural language processing systems, from spell checkers and thesaurus lexicons to machine translation.
    3. The speech engineer, who uses statistical techniques for building and evaluating the performance of speech models and language models.
    4. The professional programmer and the software engineer, who take basic specifications from a linguist or computational linguist and develops "industrial strength" applications.

6.2 The case of the Ibibio lexicon and concordance

Within the framework of two joint projects with the universities of Bielefeld and Uyo, two items of computational lexicographic research and development were carried out: preparation of a lexicon (Urua, Ekpenyong & Gibbon 2004a) and a research-orientated concordance (Urua, Ekpenyong & Gibbon 2004b). Short samples from the dictionary and the concordance are shown in Figure 3 and Figure 4. Some of the issues involved in development are outlined in this section.

When preparing the Ibibio lexicon, we were confronted by a typical situation: a "legacy print lexicon" produced with a typewriter font. After scanning the document, attempts to extract a sensible working digital document failed because of the quality of the print and the paper, and because of inconsistencies in formatting. It turned out to be easier to have the dictionary re-typed locally as a simple table database.

With the dictionary available in a systematic tabular form, it was straightforward to check for inconsistencies by re-sorting the table according to different criteria and checking for mis-spellings and other errors.

When these errors had been removed, a fairly simply programme, of a kind which any student of computer science can produce, was developed in order to format the dictionary for various purposes:

1. Extending by the addition of further lexical items.
2. Printing on paper (in a more consistent format than the original) whenever a new version makes this desirable.
3. Automatically re-formatting as a hyperlexicon for linking to the World Wide Web or putting

on a CD.

In fact, the hyperlexicon was produced before we did it, by two expatriate Ibibio software engineers in the USA who discovered our dictionary on the web (see Figure 5).

Similarly, the concordance was produced from typed journalistic texts which contained font inconsistencies which had to be ironed out by means of a word extraction and sorting programme. On the basis of a wordlist which was automatically extracted from the texts, the contexts in which the words occur in the texts was found automatically and formatted as a printable concordance.

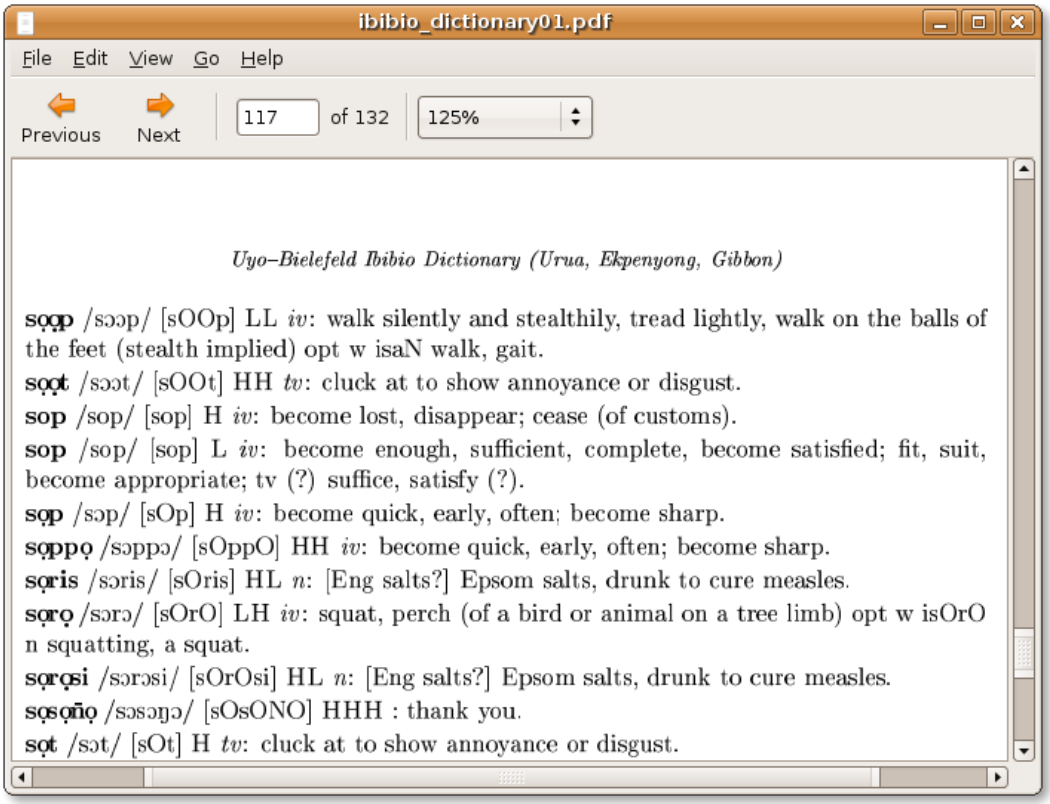

Figure 3: Ibibio dictionary excerpt.

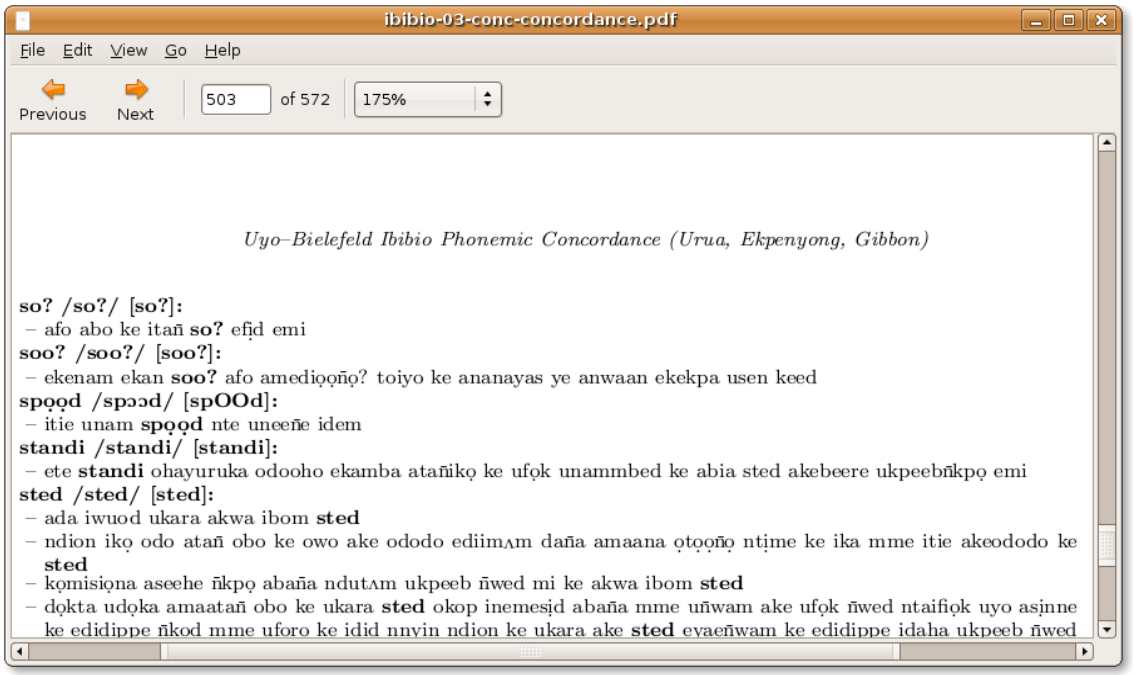

Figure 4: Ibibio concordance excerpt.

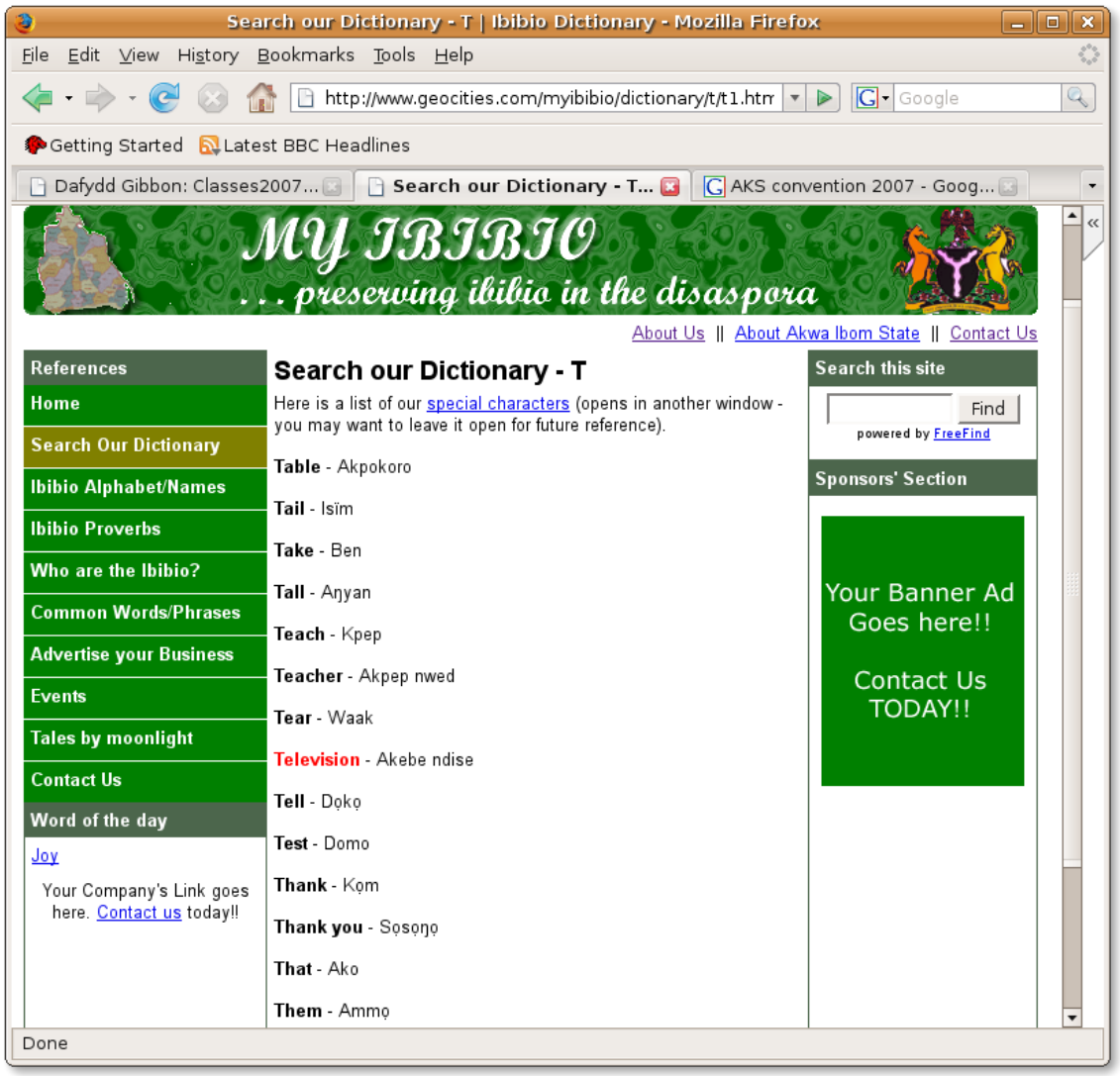

Figure 5: Hyperlexicon version of Uyo-Bielefeld Ibibio dictionary (converted by Emem Akpan and Itoro Akpan-Iquot).

6.3 The case of the Ibibio speech synthesiser

As the contribution of the Uyo and Bielefeld partners to the Local Languages Speech Technology Initiative consortium (LLSTI), a speech synthesiser was prepared by Moses Ekpenyong in cooperation with Eno-Abasi Urua, Dafydd Gibbon and Ksenia Shalonova of the host company, Outside Echo Ltd., Bristol. The technical details are not relevant here, but the linguistic contribution included the following, based on Eno-Abasi Urua's work on Ibibio phonology:

1. Creation of a formal table of grapheme-phoneme correspondences and their conditions.
2. Formalisation of phoneme cooccurrence rules.
3. Creation of a diphone table (pairwise occurrences of phonemes) by
    1. automatic conversion of the text into a phonemic transcription,
    2. automatic extraction of pairs of adjacent phonemes.

In addition, as a contribution to a future full version of the speech synthesiser with correct assignment of prosody, a complete grammar for Ibibio simple sentences was produced, including Subject-Verb and Verb-Object agreements. This grammar was formalised as a Finite State Transducer; the complexity of a Phrase Structure Grammar is not necessary for simple sentences; it has long been known that Chomsky's claim that languages are not Finite State languages is not

correct when applied to simple sentences and even some kinds of complex sentence. It was in fact straightforward to convert the relevant parts of Okon Essien's very detailed and explicit grammar of Ibibio into this formalism.

7 Conclusion: a charter for computational language documentation

Many more things could be said about why linguists should compute, particularly from the point of view of theoretical linguistics. However, this would be a topic for a whole seminar, not a conference lecture. The obvious way to stop is to formulate a programme for future collaborative work between linguistics of all persuasions, including field linguists and computational linguists. A charter for a paradigm of "Workable Efficient Language Documentation" (WELD), with the CESAF criteria mentioned at the beginning of this contribution, was already published some years ago (Gibbon 2003), and will be reviewed here in conclusion.

A Charter for the WELD paradigm would include at least the following five benchmark principles of *comprehensiveness*, efficiency, *state of the art*, *affordability* and *fairness*:

1. Language documentation must be comprehensive. In principle this means that language documentation must apply to all languages. But economy is a component of efficiency, and priorities must be set which may be hard to justify in social or political terms: if a language is more similar to a well−documented language than another language is, then the priority must be with the second.
2. Language documentation must be efficient. Simple, workable, efficient and inexpensive enabling technologies must be developed, and new applications for existing technologies created, which will empower local academic communities to multiply the human resources available for the task. A model of this kind of development is provided by the Simputer ("Simple Computer") handheld Community Digital Assistant (CDA) enterprise of the "Bangalore Seven" in India (see <http://www.simputer.org/>), which could easily be incorporated into European and US project funding.
3. Language documentation must be state−of−the−art. In addition to using modern exchange formats and compatibility enhancing archiving technologies such as XML and schema languages, efficient language documentation requires the deployment of state of the art techniques from computational linguistics, human language technologies and artificial intelligence, for instance by the use of machine learning techniques for lexicon construction and grammar induction. The SIL organisation, for example, has a long history of application of advanced computational linguistic methodologies (see <www.sil.org>), and more research is needed here.
4. Language documentation must be affordable.In order to achieve a multiplier effect, and at the same time benefit education, research and development world−wide, local conditions must be taken into account. Traditional colonial policies of presenting "white elephants" to local communities which must be expensively cared for and then rapidly become dysfunctional, must be replaced by inexpensive dissemination methods − at third world Internet prices, it can cost hundreds of Euros to download a large, modern software package (not counting landline interruptions), and net−based registration and support is costly.
5. Language documentation must be fair. If a language community shares its most valuable commodity, its language, with the rest of the world, then the human language engineering and computational linguistic communities must do likewise, and provide open source software (also to reap the other well−known potential benefits of open source software such as transparency and reliability). The Simputer Public Licence for hardware and the Gnu Public Licence for software are useful references. The development and deployment of proprietary software (and hardware for that matter) and closed websites in this topic domain is a form of exploitation which is ethically comparable to other forms of one−way exploitation in biology and geology, for example in medical ethnobotany and oil prospecting.

There are many initiatives in the area of Documentary Linguistics and related disciplines to establish criteria for good resources and tools for making documentation more effective. The SIL organisation has always used this kind of methodology, for example, but there are international conferences such as the *Language Resources and Evaluation Conference*, LREC, with which some of you are familiar, which brings together linguists and engineers. In this context, Krauwer has developed the BLARK (Basic Language Resource Kit) set of toolkit specifications (Krauwer 2003[1]). A final example, one of the most recent ones, is the OLAC initative, the *Open Language Archives Community*, initiated by Gary Simon of SIL, and by Steven Bird of the University of Melbourne, who combines expertise in computational linguistics with extensive fieldwork experience in Cameroon.

Concepts for training in the computational documentary linguistics paradigm have already been developed (Gibbon & Borchardt 2007; Urua, Ekpenyong, Gibbon & Ahoua 2006). One way to intensify cooperation within this paradigm, and at the same time to save financial resources which arise from travelling, is "tele-lecturing" via Skype (or other "Voice over IP", VoIP, i.e. internet telephony, providers): I arranged for this to be done this twice, each time successfully, once with a group of students in Berlin (last January, when a hurricane stopped all trains in Germany and prevented me from travelling), and once for a workshop in Addis Ababa which I was unable to attend for financial reasons. Each of these lectures cost precisely nothing in addition to the basic cost of the internet connection; since broadband internet connections are generally charged at a flat rate and not by time or volume, no extra cost was incurred. This could, in the medium and long term, be a viable and economical model for our own further cooperation.

But first of all, the infrastructures must be established: this is a political and economic issue. Applications need to be developed (cf. the model for geographical information systems in Ekpenyong, Umoh, Udoinyang, Ibioang, Urua & Gibbon 2006; Ekpenyong, Urua & Gibbon 2004). Support needs to be argued and lobbied for in concrete and persistent detail in local and national governments as an essential condition for local and national development. And the infrastructures have many other benefits outside academia – many businesses have such reliable infrastructures, and it is time for academia to participate.

In this sense, I hope that we will be able to contine to work together across the continents, documenting our cultures and languages for the preservation of our human heritage and for the improvement of social conditions by the deployment of speech and language technologies wherever they might be helpful.

8 References

Amsalu, Saba & Dafydd Gibbon (2005). Finite State Morphological Analysis of Amharic. In: *Proceedings of "Recent Advances in Natural Language Processing"*. Sophia, Bulgaria.

Ekpenyong, M.; Urua, E. & Gibbon, D. (2004). Local e-Government Text-To-Speech: A Speech Technology Initiative. Journal of Computer Science & Its Applications. 10(2): 125-131.

Ekpenyong, Moses, Nnamso Umoh, Mfon Udoinyang, Glory Ibiang, Eno-Abasi Urua, Dafydd Gibbon (2006). Infrastructure to Empowerment: An OSWA+GIS Model for Documenting Local Languages. In E-MELD Workshop Proceedings, East Lansing, USA.

Gibbon, Dafydd (2003). Computational linguistics in the Workable Efficient Language Documentation Paradigm. In: Gerd Willée, Bernhard Schröder & Hans-Christian Schmitz, Ed.. (2003). *Computerlinguistik: Was geht, was kommt?* St. Augustin: Gardez! Verlag, 75-80.

Gibbon, D.; Urua, E-A. & Ekpenyong, M. (2004). Data Creation for Ibibio Speech Synthesis. Local Language Speech Technology Initiative (LLSTI) Publication archive. http://www.llsti.org/pubs/Ibibio_data.pdf.

Gibbon, Dafydd and Eno-Abasi Urua (2006). Morphotonology for TTS in Niger-Congo languages. In: *Proceedings of 2rd International Conference on Speech Prosody*. Dresden: TUD Press.

Gibbon, Dafydd, Eno-Abasi Urua & Moses Ekpenyong (2006). Problems and solutions in African

1 See also http://www.elda.org/blark/

tone language Text-To-Speech. In: Justus Roux, ed., *Proceedings of the Multiling 2006 Conference*, Stellenbosch, South Africa.

Gibbon, Dafydd & Nadine Borchardt (2007). Computational lexicography: a training programme for language documentation in West Africa. In: B.M. Mbah and E.E. Mbah, eds., *Linguistics in History: Essays in honour of P.A. Nwachukwu.* Nsukka, Nigeria: University of Nigeria Press.

Krauwer, Steven. (2003). The Basic Language Resource Kit (BLARK) as the First Milestone for the Language Resources Roadmap. In: *Proceedings of SPECOM 2003,* Moscow.

Urua, Eno-Abasi, Moses Ekpenyong, Dafydd Gibbon, Firmin Ahoua (2007). Developing a Master's Programme on Language Documentation for Local Languages. *Proceedings of UNESCO/ACALAN Conference Identifying Good Practices in Safeguarding Endangered Languages*, Addis Ababa, Ethiopia February 9-10, 2007.

Urua, Eno-Abasi, Moses Ekpenyong & Dafydd Gibbon (2004a). *Uyo Ibibio Lexicon*. ABUILD Language Documentation Curriculum Materials.

Urua, Eno-Abasi, Moses Ekpenyong & Dafydd Gibbon (2004a). *Uyo Ibibio Concordance*. ABUILD Language Documentation Curriculum Materials.