Annotation Driven Concordancing: the PAX Toolkit

Thorsten Trippel, Dafydd Gibbon

Department of Linguistics and Literary Studies
Bielefeld University
Postf. 100131
33501 Bielefeld
Germany
{ttrippel, gibbon}@spectrum.uni-bielefeld.de

Abstract

We describe PAX, "Portable Audio Concordance System", a proof-of-concept prototype of a multipurpose, multilingual audio concordance toolkit. The primary goal is to support efficient grammar and lexicon construction in the documentation of unwritten languages; languages currently included are Ega, Anyi, and Koulango (Ivory Coast), additional samples in German and English. The approach combines methods from corpus linguistics, annotation theory and practice, phonetics and lexicography.

1. Objectives

Finding occurences of selected utterances in multimodal corpora for multimodal lexica is the objective of the *Portable Audio Concordance System* (PAX)[1].

Modern dictionaries these days claim to be corpus based, for example lexica from the COBUILD project (see (Sinclair, 1987)). This is in the sense that

1. the order of different meanings corresponds to the frequency in a defined corpus
2. the examples for the use of different words are taken from *real world* data, i.e. corpora.

This presupposes a sufficiently preprocessed (marked-up) textual source. For written texts there are a number of corpora used for this purpose such as the *British National Corpus* (British National Corpus, 2001) for English or the copora available via (COSMAS, 2002) for German.

However, these corpora contain written texts, and there are concordances for lexical analysis of written texts, which are well known (see for example (van Eynde and Gibbon, 2000)), but no adequate concordancing tools for spoken language exist. The concordancing task for spoken language is difficult: units are less well identified, access to both transcription text and speech signal is required, and standard aids like word statistics need to be supplemented by visualised transformations of the speech signal.

We demonstrate an enhanced *KeyWord in Context* (KWIC) concordance, based on a search space as defined by the annotation graph (Bird and Liberman, 2001), representing the transcription, and a search, which includes a variety of complex criteria. The XML formalism is based on the TASX format as described by (Milde and Gut, 2001).

The position of a concordance in a concordance based multimodal lexicon system is described by Figure 1. Starting from the annotation of a multimodal source a lexicon is generated that falls back onto the annotation via the concordance for exemplified usages and possibly for evaluation of generated lexicon entries. The annotation itself is used by the concordance as data input as well, and the annotation can be refined within fixed environments by using a concordance. Hence there is a bidirectional connection between the concordance and annotation.

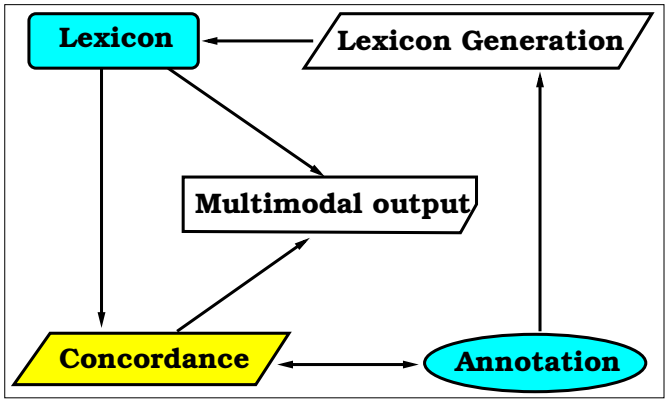

Figure 1: Concordances in a multimodal Lexicon system

1.1. Methodology

The PAX concordance design is based on a function

$$f : CORPUS \rightarrow < KWIC, SIGTRANS >$$

where $CORPUS$ is a set of annotated signals partitioned for different languages, KWIC is the keyword in context concordance, and $SIGTRANS$ is a set of signal output renderings (audio, waveform, F_0, spectrogram). The corpus consists of digital signals, which were annotated at different levels.

The process of concordance generation consists of four functions, namely:

1. f_{lexSL} lexicon generation function for spoken language dictionaries

$$f_{lexSL} : Corpus \rightarrow Lexicon$$

2. f_{anno} annotation generation function

$$f_{anno} : Signal \rightarrow Annotation$$

[1] The acronym is derived from PACS by merging the final letters.

3. f^{-1}_{lexSL} annotation access function

$$f^{-1}_{lexSL} : Lexicon \rightarrow Annotation$$

4. f^{-1}_{anno} signal access function

$$f^{-1}_{anno} : Annotation \rightarrow Signal$$

The normalisation preprocessing function $f_{trans} : Annotation[org] \rightarrow Annotation[trans]$ is omitted here; it would be necessary to have this function in order to handle annotations produced by different annotators and different annotation tools and conventions. It would denote the transformation of a source annotation format into a format that can be accessed by f^{-1}_{anno} and f^{-1}_{lexSL}. However, the functions above presuppose a normalised data format.

The simplest of this lexicon functions can be seen as a generated wordlist with the latent property of each *word* being *in the corpus*. The formal lexicon model respects but is not restricted to other lexicon models based on form, meaning or use; especially there are no semantic limits as for example discussed in (Sharoff, 2002).

1.2. Concordance Use in Stand-off Annotation

In the process of first approaching and annotating data it is not uncommon to omit features that do not seem appropriate to tag or that are irrelevant for the present research task. In a later stage of reusing the corpus and the annotations, other features might become relevant (without modifying the original, see (McKelvie and Thompson, 1997)) and consequently need to be checked in correspondence with the original —which means in the context of spoken language with the recorded signal.

By using existing annotations it should be easy to access only the relevant parts of the signal for reannotation or further — not necessarily automatized — detailed analysis. In this context it is meant to use the concordance for the preselection of data.

All that is necessary would be the inverse corpus based lexion function:

$$f^{-1}_{lexSL} : Lexicon \rightarrow Annotation$$

and a function

$$f^{-1}_{anno} : Annotation \rightarrow Signal$$

An audio concordance needs to provide these two functions. If possible further functionality should be added, such as interfaces to phonetic analysis software for automatic processing of selected parts of signals and basic corpus statistical features, such as word counting and word frequency analysis, also *Type-Token-Ratio* (TTR) can be added.

1.3. Funtional Requirement Specification

The functionality of the concordance system sets certain technical and functional requirements for the implementation. Additionally it should respect our general requirements for PAX:

Standardisation: For coding standard requirements are used in order to avoid incompatibilities and to promote exchangeability. Proprietory — in the sense of not openly accessible and usable — formats are strongly discouraged.

Interoperability: Support for major platforms such as Unix/Linux, Windows, Macintosh should be aimed at, because these platforms are frequently used by field workers and linguists.

Multifunctionality: The system should be extensible to different requirements and new functionality as needed by the community.

Low-cost/low-end, online/offline: As the target user group includes institutions and persons in areas without access to the newest IT infrastructure, the system should be independent of recent software versions or high-end hardware in order to ensure usability under local conditions of this kind. As networking devices might not be available in fieldwork locations, offline functionality is needed

Network Access: For training purposes and for the consistency of data, network access is to be provided.

2. Design

Our design strategy is to use a KWIC (KeyWord In Context) approach, taking an annotated database of speech signals as input, with standard typewriter-friendly SAMPA transcription, XML annotation formats, and a suite of format converters to cope with data input from different corpora, or annotators with different software and hardware platforms. Conceptually the approach is not very different from statistical training procedures used in spoken language technology, but the requirements are very different in detail.

Figure 2 shows a detailed overview on the PAX concordancing systems design.

The PAX architecture is modular, with wordlist extraction and KWIC concordance construction modules (Perl), and signal extraction and processing modules (Java programs and Praat scripts). These modules feed three independent user interfaces:

The system consists of three basic modules:

Data acquisition module: The data acquisition module calculates corpus information, based on available corpora in specified locations. Among other information a wordlist is collected, a list of available annotation tiers and subcorpora. The search procedure uses this information as a basis for further processing, involving the generation of a static (predefined and accessible) and dynamic (on the fly generated) concordances.

Corpus consultation module: The user selects search criteria from the information provided by the data acquisition module and defines an output filter to specify the size of the context (e.g. the number of words or characters left and right of the keyword occurrence). The output of the module is the KWIC concordance. Each

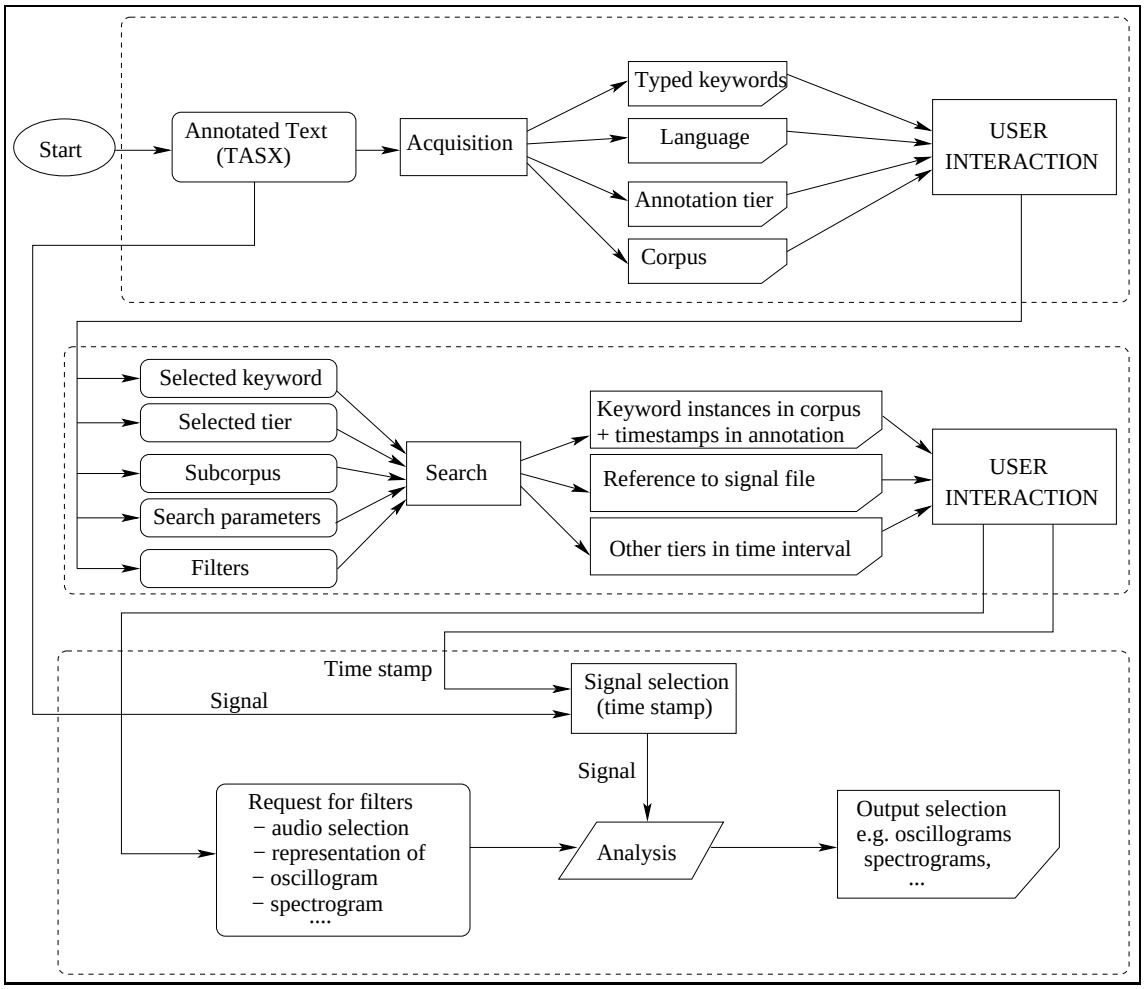

Figure 2: Design of *PAX*, based on TASX corpus format

KWIC context is supplemented with a further selection panel specifying a set of choices for access to the signal: a range within the context; waveform or spectrogramme or pitch track; selection of further output in the same window or a new window.

The consultation module provides some basic statistical information as well. Based on the distribution function measures of variability are given (Oakes, 1998), such as mean, range and median, standard deviation, type/token ratio (for simple tokens). The number ratio of matches is also given. However, the interpretation of these measures is left to the user as it is dependent on the data types whether a measures is relevant or not.

Signal analysis: The KWIC output contains selection panels for further processing on the signal level; segments are selected on the level of time-stamps assigned to the selected context. These segments are passed to signal processing for further analysis.

3. Implementation

Input data are time-aligned in SAMPA standard ASCII IPA coding (modified for efficient tone language coding), using Praat, Transcriber or esps/waves+. These operational formats are converted to an XML annotation graph format (Bird and Liberman, 2001) the TASX DTD (Milde and Gut, 2001), retaining SAMPA coding. TASX-XML and SAMPA fulfil the archive exchange and low-end availability requirements. The TASX DTD is available at: `coli.lili.uni-bielefeld.de/~milde/tasx/`

The wordlist extraction, KWIC concordance construction modules and format converters are in Perl, signal extraction and processing modules are in Praat scripts and Java. These modules feed three independent GUI modules, see section 3.2..

This hybrid implementation strategy fulfils the low-cost, low-end, on/offline and interoperability requirements.

3.1. Pragmatic Choice of Programming Language

PAX is implemented with a hybrid component structure, and the programming languages for the components were selected on a pragmatic basis:

Perl: the modules itself are implemented in Perl, as it is available for many computer systems (including UNIX/Linux systems, MS-Windows, MacOS) and resource friendly. Additionally it provides a rich source of regular expression capabilities and many interface format libraries, covering command line access, graphical user interfaces and CGI-access.

Perl allows access to other system components as well

and can be used to call other modules from within the program.

Java: large-scale signal processing is impossible with traditional scripting languages such as Perl. For signal processing we selected Java as a suitable language that is system independent. The problem of Java being relatively resource unfriendly could be neglected for the present implementation as Java plays a minor role in the toolkit and does not result in a bottle-neck in performance. On the other hand no comparable system independence could be accomplished by using other programming languages.

Praat: for signal processing the routines of the Praat programme for enhancing phonetic productivity are used. These are easy to incorporate since Praat provides a scripting language and interface that can be invoked by other programs. Praat is also available for all major platforms.

R: the statistical functions provided by the R-statistics package are connected to the concordance for effective availability of various statistic functions. R was chosen because it is freely available under GNU public licence and because it is available for all major platforms (Gentleman and Ihaka, 1997 2002).

3.2. Select the User interface

The PAX implementation exists with different user interfaces, all built upon the same core algorithms; they correspond to the design specification, including wordlist extraction and KWIC concordance modules in Perl, signal extraction and processing. They are:

1. a CGI application with HTML and WAV output for use with a web server
2. a TK application for offline use and without a local www server
3. a low end command line based access, also used as an interface for other programs.

3.3. Interface Structure

The interfaces are clearly structured for easy access.

Enter the concordance system by selecting a language. This directs the system to the designated location, holding a corpus for a specific language in the TASX format.

Selecting a keyword in a subcorpus and tier is possible in the next interface. At the present stage the keyword, subcorpus and tier can be selected independently from each other so the result is only related to the selected subcorpus and tiers. Additionally the user can select the environment by a specifying a numerical values of words before or behind the keyword.

Resulting contexts are presented with numberous additional options on the signal analysis containing the word and a further specified context. Additional statistical information is produced and presented here as well in a simple table form. A sample interface of results is shown in Figure 3.

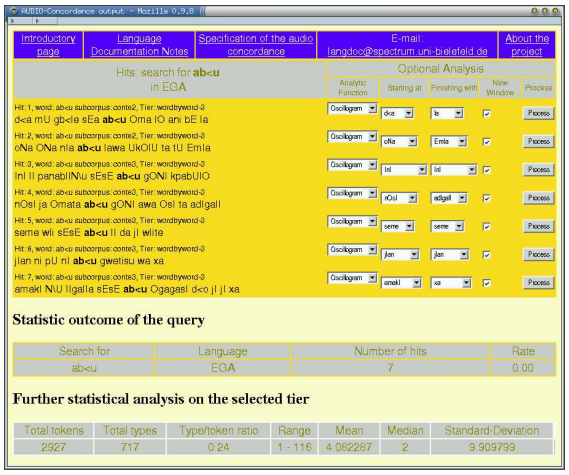

Figure 3: Keywords in context with additional options for further signal analysis; corpus statistics at the bottom of the page

Analytic functionality is achieved by the use of Praat, producing appropriate images and value tables.

4. Evaluation

The toolkit was evaluated following EAGLES guidelines as defined in (Gibbon et al., 1997), with

1. inhouse testing for correctness of results,
2. in-project testing with respect to substantive and ergonomic user requirements,
3. extension to quite different corpora, including the VerbMobil German speech database and a German-English language acquisition corpus.

5. Summary and Further Development

The PAX tool was developed specifically as part of an environment for efficiently analysing spoken language, in particular unwritten languages, including African tone language annotations with tone markup. The specifications for the tool were established on the basis of experience in previous work on encyclopedia modelling for African languages funded by the Deutscher Akademischer Austauschdienst, work on the efficient analysis of endangered languages funded by the Volkswagen Foundation, and work on the construction of multimodal lexica funded by the Deutsche Forschungsgemeinschaft. At the present time the PAX application contains corpora from five languages, three West African tone languages (Anyi, Ega, Koulango) and two European languages (English, German).

Current work is directed towards tagging enhancement, modules for further spoken language corpus analysis, and time-aligned multimodal data.

The next step will include the use of the concordance in a multimodal lexicon system of spoken language data. The concordance will be used for creating real-world examples for lexicon entries.

6. References

Steven Bird and Mark Liberman. 2001. A formal framework for linguistic annotation. *Speech Communication*, (33 (1,2)):23–60.

British National Corpus. 2001. British national corpus. CD-Rom.

COSMAS. 2002. Corpus storage, maintenance and access system. http://corpora.ids-mannheim.de/ cosmas/.

Robert Gentleman and Ross Ihaka, 1997 - 2002. *The R Project for Statistical Computing*. http://www.r-project.org.

Dafydd Gibbon, Roger Moore, and Richard Winski, editors. 1997. *Handbook of Standards and Resources for Spoken Language Systems*. Mouton de Gruyter, Berlin.

David McKelvie and Henry S. Thompson. 1997. Hyperlink semantics for standoff markup of read-only documents. In *Proceedings of SGML Europe'97*, Barcelona, May.

Jan-Torsten Milde and Ulrike Gut. 2001. The tasx-engine: an xml-based corpus database for time aligned language data. In *Proceedings of the IRCS Workshop on Linguistic Databases*, Philadelphia. University of Pennsylvania.

Michael P. Oakes. 1998. *Statistics for Corpus Linguistics*. Edinburgh Textbooks in Empirical Linguistics. Edinburgh University Press, Edinburgh.

Serge Sharoff. 2002. Meaning as use: exploitation of aligned corpora for the contrastive study of lexical semantics. In *Proceedings of LREC 2002*, volume this volume, Las Palmas.

J. M. Sinclair, editor. 1987. *Looking up*. Collins ELT, London.

Frank van Eynde and Dafydd Gibbon. 2000. *Lexicon Development for Speech and Language Processing*. Kluwer Academic Publishers, Dordrecht.