Spock - a Spoken Corpus Client

Maarten Janssen, Tiago Freitas

IULA/ILTEC, ILTEC
Plaça de la Mercé 10-12 Barcelona, Rua Conde de Redondo 74-5 Lisboa
maarten@iltec.pt, taf@iltec.pt

Abstract

Spock is an open source tool for the easy deployment of time-aligned corpora. It is fully web-based, and has very limited server-side requirements. It allows the end-user to search the corpus in a text-driven manner, obtaining both the transcription and the corresponding sound fragment in the result page. Spock has an administration environment to help manage the sound files and their respective transcription files, and also provides statistical data about the files at hand. Spock uses a proprietary file format for storing the alignment data but the integrated admin environment allows you to import files from a number of common file formats. Spock is not intended as a transcriber program: it is not meant as an alternative to programs such as ELAN, Wavesurfer, or Transcriber, but rather to make corpora created with these tools easily available on line. For the end user, Spock provides a very easy way of accessing spoken corpora, without the need of installing any special software, which might make time-aligned corpora corpora accessible to a large group of users who might otherwise never look at them.

1. Introduction

Time-aligned multimodal corpora are an important tool in the study of spoken language nowadays. These are corpora that consist not only of a sound file, but also the (orthographic/phonetic) transcription of the sound file, as well as an alignment table indicating which fragment of the sound file a given sentence or phrase in the transcription file corresponds to.

There are various tools around for the creation of time-aligned corpora, such as WaveSurfer (Sjölander and Beskow, 2000), ELAN (Brugman and Russell, 2004), PRAAT (Boersma, 2001), WinPitch (Martin, 2003), and Transcriber (Barras et al., 2001). Many of these tools provide a rich set of features to explore both the sound file and the transcription, and most of them are free and readily available.

However, the distribution of spoken corpora using these programs has several restrictions: sound files of any significant spoken corpus are usually enormous - easily surpassing 50Gb. The distribution of files of that size is not feasible over the Internet or any other easy means of transport, and has to rely on large storage media such as DVD, meaning they are not immediately available but have to be shipped physically. Obtaining the files of the spoken corpus is not sufficient to work with them: it is necessary for the person who wants to consult the corpus to have the program with which it was created installed on his computer (or a similar program with import facilities). Given that most of these applications have many options for both creating and exploiting the corpora, they are not always easy to handle for a first-time user.

Because of these restrictions, spoken corpora are not as readily available as they might be, and certainly not easily accessible for occasional users. Spock is a lightweight application that attempts to resolve this issue by providing easy, online access to time-aligned corpora. The main purpose of the tool is to make time-aligned corpora accessible to a large group of users that might otherwise never look at them.

2. Spock - Overview

Spock is a web-based application for the exploitation of time-aligned corpora. Contrary to other available online tools, such as ANNEX (Berck & Russel, 2006), Spock is a transcription-driven rather than a sound-oriented spoken corpus tool.

Spock is fundamentally a concordancer tool that provides the corresponding sound files for all the results. The idea behind this is that although spoken corpora are primarily valuable because of their sound data, the selection of fragments is difficult, and most often best provided via a search on the transcription: it is easy to select relevant sound fragments based on orthographic clues, whereas it is very hard to find fragments based on their audio characteristics.

Spock was developed for the CORP-ORAL project (Freitas and Santos, 2008), a spoken corpus of spontaneous speech of European Portuguese, transcribed using ELAN. Although some of the tools of Spock are dedicated to the ELAN file format and the transcription norms used in the CORP-ORAL project, the system itself works with any type of transcription any a number of transcription formats. Since only the data of the CORP-ORAL project are available for the moment, all examples in this paper will be taken from that corpus. The CORP-ORAL corpus can be queried with the Spock system at the following URL: `http://www.iltec.pt/spock`

2.1. Audio Concordancer

The main function of Spock is keyword concordancing on time-aligned corpora. Spock allows you to look for words or sequences of words in the transcription of a spoken corpus, in the same way that standard concordancers for written corpora, such as MonoConc (Barlow, 2000) do. When querying for a word or phrase, it gives a list of all the contexts in which the word appears. The difference with a traditional concordancer is that Spock not only gives the orthographic contexts, but also the audio context for each of the matches. An example of a query for the word *euros* on the CORP-ORAL corpus is shown in figure 1. Each

line shows a matching context, with the word *euros* highlighted. In front of every context line, Spock provides a button, which will play the audio fragment corresponding to the context behind it.

Figure 1: Spock screenshot with result for *euros*

The context shown in each line is the line in the transcript file containing the query word. Since the data are transcription data, each line will typically contain not a sentence (as in a traditional concordancer), but rather a prosodic unit, that is to say a stretch of transcribed conversation between two prosodic tiers.

For each context in the result list, Spock provides a link to a page containing a larger stretch of the context of that match. The context page provides not only the matching prosodic element itself, but also a number of elements before and after it, as well as the complete sound fragment of that larger context (see figure 2). This makes it possible to look at and listen to the larger context of the matching element, in order to, for instance, disambiguate a word or sentence, or compare the pitch, tone, and speech of the element to the surrounding discourse units.

Figure 2: Screenshot of the Context screen

This very simple and straightforward way of searching time-aligned corpora provided by Spock makes it very easy for users to search for phonetic or phonological phenomena based on their orthographic realization. The next section describes two cases in which Spock can be of great use, providing real corpus data to study phonetic differences. Without a tool like Spock to study time-aligned corpora, finding examples in these cases is a tedious process of going through long hours of recordings. If more corpora are made available to the linguistic community by the means of a concordancer like Spock, data driven phonological research of the type described below becomes much easier, creating the potential for significant advances in the study of spoken language.

2.1.1. Example Queries

One example of the usefulness of Spock for conducting phonological research is the following. There are at least two ways of contracting the words *com* (with) and *que* (that) with the determiner behind it, the difference between which is socially marked. The standard pronunciation has a glide in it, and is relatively close to the orthography. The non-gliding contraction of the two words is supposedly less prestigious - in chat forums, you often find the two words even graphically contracted to, for example, *ku* for *com o*. According to Vigário (2003), there are various factors determining the final realization of such sequences with *com*, including the category of the subsequent word and the segmental context. Using Spock, it is very easy to find all occurrences of *com/que* + *o/a* in the CORP-ORAL corpus. In the original transcriptions, all occurrences are transcribed as separate words, which means that it is necessary to be able to listen to the sound file in order to verify the actual realization. The fact that Spock provides the WAV file next to the transcription allows you to quickly determine for each occurrence if the two words are contracted or not, which in turn gives easy access to the data necessary to further analyze exactly which factors play a role in this.

Another example of the use of Spock is the following. As has been observed by various authors, the vowel quality in neoclassical compound words is highly variable. For the same word, there can be several concurrent pronunciations of the vowels in the neoclassical part of the word. For instance, for the word *economia* (economy), you can find up to nine different pronunciations: the *e* can be pronounced as [e], or as [ɛ], or even as [i], and the *o* can be pronounced as either [o], [ɔ], or [u]. The full Cartesian product of these options can be encountered - from [ekonuˈmiɐ] to [ikunuˈmiɐ]. To study the relative frequency of these realizations, as well as trying to establish which factors play a role in the selection of one of them, it is very convenient to search for all words in the CORP-ORAL corpus that start with *eco-*.

2.2. Server-Side Chunking

In stand-alone time-aligned corpus tools, it is possible to listen to a specific bit of the spoken corpus by simply jumping to the desired time index in the sound file. This is possible because the sound file is stored locally on the user's desktop computer and can be easily accessed. In an Internet based spoken corpus client that option is not available:

in order to be able to jump to a given time-index in the sound file, it would first be necessary to transfer the entire sound file. In order to be feasibly listen to bits of the sound file in an online system, it has to be cut into small fragments - most logically to those bits that correspond to the prosodic units. The average prosodic unit is typically only some seconds long, and the corresponding sound fragment only around 100kb, a size which can quickly be transferred from a web-page.
The easiest way to split up the sound file would be to use an audio program to divide the sound file into the chunks corresponding to each of the prosodic units, and to store all the sound chunks individually on the server. However, in many cases it is useful to be able to listen to just a bit before or after the actual prosodic unit, for instance to listen to transitions between words. If the sound file is chunked up, it would only be possible to listen to the neighbouring sound chunks, which might just miss the relevant transition points.
Therefore, all sound chunking in Spock is done on-the-fly: whenever a sound fragment is needed, the desired fragment of the sound file is created temporarily, and transferred to the user. In this way, it becomes possible to extend the sound fragments by several seconds, or to listen to the sound fragment corresponding to a sequence of several prosodic elements in a row.

2.3. SimpleConc and YakwaSI

The concordancer system used in Spock is based on a lightweight open-source concordancer called SimpleConc, developed at ILTEC in Lisbon, which in turn was based on the YakwaSI system developed at the ERSS in Toulouse. SimpleConc can be used independently of Spock for written corpora, and can be obtained from the ILTEC web site (`www.iltec.pt`). SimpleConc provides a simple, standard set of concondancer search options for plain text corpora: you can look for a full word or for sequences of words. To look for parts of words, you can introduce a wildcard character - the asterix (*) matches zero or more characters, whereas the plus (+) requires at least one character in that place. Here are some example queries in the SimpleConc system:

- **word** will look for all occurrences of the word *word* in the text
- **wor+** will look all words that start with *wor* and have at least one letter after that
- **w*ord** will look all words that start with a *w* and end on *ord*, including the word *word*
- **+ly a*** will look all words that end with *ly*, that are followed by a word starting with an *a*

It is possible to use Spock for querying texts that have been morphosyntactically tagged. For POS tagged corpora, you can specify for each word in the search query which morphosyntactic class it should belong to. It is possible to either simply look for any word that is say a noun or verb, or to look for specific words or parts of words of a given class. Below are some example queries of the enriched query system with POS tagged corpora. The POS tag *Noun* in these queries is just used for clarify - the actual tag for the morphosyntactic class depends on the tagset of the POS tagger used.

- **the Noun** will look for all occurrences of the word *the* followed by a noun
- **hammer:Verb** will look for all occurrences of the word *hammer* that are classified as verbs, ignoring the nominal occurrences
- **+ly:Noun** will look for all nouns that end on *ly*

All these query options of SimpleConc are available in Spock – with the addition that Spock provides the audio context for each of the matching phrases (prosodic units). However, there are some other features in SimpleConc that were not ported to Spock, since they were considered less useful with respect to the sound-oriented queries provided by Spock. For instance, SimpleConc can give a frequency ordered list of all the words in the corpus. It also has a context count option, which for any given search query provides a list of the words most commonly occurring to the left and right of it, order by frequency. And SimpleConc provides some different display options, including the traditional view with the search query centred in bold-face, and a fixed amount of context to each side. Although it would be easy to add these functions to Spock, they have not currently been implemented for Spock.

2.4. Speaker Selection and Privacy

For various types of queries, it is useful to be able to restrict the corpus based on the characteristics of the speaker. For instance, it would be useful to be able to look only at utterances by female speakers, or by speakers under the age of thirty. In the Spock transcription files, every line is marked with an ID of the speaker of that particular prosodic unit. In a separate file, known characteristics of each speaker are stored, such as gender, age, geographical area, and education level. In the advanced search section, it is possible to restrict the match to only speakers of a given gender or geographical area, and below or above a certain age or education level.
Although the database with the ID of the speakers also provides the name of the speaker, that information is not displayed due to privacy issues. For that same reason, it is possible to mark all occurrences of person names in the transcription file with a tag, identifying them as sensitive data. All strings marked as a person name in the transcription are not displayed in the query results in Spock, but rather replaced by the filler "*proper name*". This to further protect the privacy of the speakers. Of course, this privacy protection is only limited, since the sound file will still contain the full name of the person mentioned, unless the an audio editor is used to mask the corresponding sound as well. But this simple trick makes it impossible for the proper names in the corpus to show up in Google or other search engines.

2.5. Phonetic Transcription Search

In many cases, it will be useful to look for spoken data based on how they are pronounced, independently of how they are written. For example, for English it would be useful to be able to look for all words that end in [ʌf], independently of whether they are written with *-ough*, as in the case of *rough*, or *-uff* as in *bluff*. Although the relation between orthography and pronunciation in Portuguese is more regular than in English, even for Portuguese it is very useful to be able to search for prosodic units by means of IPA symbols.

In principle, the transcription in Spock can be of any type - including a transcription in IPA. By having a file in the Spock file format where each line contains the phonetic transcription of a segment of the sound file, it will be possible to work with Spock using either IPA or SAMPA in the same way as it is possible to work with an orthographic transcription. However, there are two ways in which Spock provides a more dedicated way of working with phonetic transcription files. Firstly, Spock comes with an IPA input box, which makes it a lot easier to type in phonetic symbols online - and it can translate queries and results back and forth between SAMPA and IPA.

Secondly, most time-alignments for phonetic transcription align individual segments to their time-index. That means that every line in the transcription file contains only one phonetic symbol. This makes it very difficult to look for sequences of symbols, which is what the user most typically wants to look for. Since Spock does not provide acoustic analysis, the precise time-index of the segement is not that relevant. It is more useful to have all the symbols for a prosodic unit assembled together on a single line. The administration environment of Spock allows you to generate such "flattened" transcription files based on joining the orthograpic and the phonetic tier of the transcription file. Both of these phonetic features in Spock are currently still in beta-phase, but should be available soon.

3. Specifications and Features

3.1. Comparison

Spock is far from the first tool available for dealing with time-aligned corpora, and (deliberately) not the most feature-riddled by a long shot. This section presents a brief comparison between Spock and some alternative tools for working with time-aligned corpora.

Contrary to Praat or ELAN, Spock does not provide an advanced text insertion interface or precise acoustic analysis tools. It does not permit the user to link annotations to audio or video data, or to establish relationships between annotations. Spock was never meant to be a program for creating time-aligned corpora (which cannot be feasibly done in an online manner), but only intended to make corpora created with other tools more easily available.

Spock does display the transcription, and allow playing the sound file, but does not provide a graphical analysis of the sound file. The typical waveform display of the audio stream, and especially more elaborate types of visual information such as F0 plots and formant tracking are completely absent from the system. For users interested in getting detailed acoustic information like that, it will be necessary to download the sound file, and open it in the program of their choice. The heavy calculations, as well as the interactive way in which such information should be displayed, make this type of information not very fit for online display.

In a sense, Spock is most comparable with TASX (Milde and Thies, 2002) and Audiamus (Thieberger, 2007) - both are intended as concordancers over transcribed corpora. However, both of these extract the text from time-aligned corpora to provide a rich set of concondance tools, which makes it impossible to listen to the actual speech related to the text. Spock features a simpler concordance tool, but maintains the relation with the actual sounds, which allows you to listen to the speech instead of merely looking at the transcription.

Many of the modern time-aligned corpus tools, such as NITE XML (Carletta et al., 2003) and ELAN, are capable of dealing with video files, having part of the screen dedicated to viewing the video material. Spock on the other hand only works with audio files. There are two reasons for this restriction - firstly, the CORP-ORAL project for which the program was developed does not work with video, and hence there was little need to build video capabilities. But secondly, the chunking techniques used in Spock are not (currently) possible for video. Chunking video is too computation-intensive, and there are no easy ways of extracting part of a video file from the command line. But even if it were possible, small video fragments are still too large to stream on a web page the way this is done with the audio files in Spock, and most servers would not allow enough disk space to host full-size video corpora in the first place.

3.2. System Requirements

Spock is an open source tool that will run on almost any UNIX or LINUX based web server. It is written in a combination of PHP and Perl, and is easy to install. The only required installation is that of an open source sound managing tool called SoX (Sound exchange), which is the software used for the sound chunking. SoX is part of the basic port system of most operating systems and should be readily available to anyone.

Since the URL and the full path to the files of the Spock system will be different for each server, there is a global settings file in which such parameters can be set. That same settings file can also be used to change some display options, as well as set the password for the administration environment described in section 3.4.

3.3. File formats

There are several standard formats for time-aligned corpora: the proprietary formats of for instance ELAN, Transcriber, Shoebox, and PRAAT, and more general formats such as the Exmaralda exchange format, as well as movie-subtitle formats such as SRT. Despite the very different design of these formats, their basic principle is the same: the sound and the annotation are stored in separate files, and the annotation file contains timestamps, referring to where in the sound file the beginning or end of a sentence of the transcription file can be found, expressed in terms of the

number of seconds (or bits) counting from the start of the sound file.

For speed optimization, Spock does not use any of these existsing formats, but uses a proprietary format for its annotation file. The format in which the annotation is stored for Spock is in a plain tab-separated text file, in which every line fully describes a prosodic unit, defining the name of the sound file the line corresponds the, the start and end times of the line, the ID of the speaker, and the (orthographic) transcription of the prosodic unit. In order to be compatible with other programs however, Spock provides import functions for files stored in ELAN, Shoebox, Exmaralda, and SRT format. Although there are more formats around, most other programs are capable of exporting their data in at least one of these formats, and ELAN can import several other file format, including the PRAAT textGrid files. In practice, Spock should therefore be usable in combination with almost any transcriber package.

The default character encoding in Spock is ISO 8859-1, since that is most compatible with standard UNIX distributions. However, the system is compatible with other character encodings as well - it has been tested with UTF-8, and should work with UTF-16 as well. This means that in principle, Spock text files can be in any language, or even in IPA encoding if so desired.

For the format of the sound files, Spock in principle has few restrictions: playing the sound file is taken care of by the internet browser of the user, which nowadays play almost any sound file. However, sound chunking is done on-the-fly with the SoX software package, meaning that real support is only provided for sound formats that are supported by SoX. In its most recent versions, SoX supports MP3 files, as well as several other popular sound formats. However, the most reliable results are obtained by using uncompressed WAV files. Bit rate and sample rate can freely be chosen, but both server requirements (in terms of disk space needed) and download times go down with smaller files. To keep to the smallest file size without significant quality loss for spoken text, CORP-ORAL uses a file format standard of 11kHz/16bits.

Spock stores the name of the audio file in every single line. Apart from an advantage in processing speed, this allows the assignment of a different sound file to each line. This is useful in case the transcription is based on a multi-channel sound file with a different channel for each speaker. If the different channels are stored in separate sound files, Spock can provide the sound of only the relevant channel. This makes the speech much more audible in cases in which the different speakers are speaking at the same time. In the presentation of the larger context, the different channels can optionally be mixed back into a single stereo file.

3.4. Administration Environment

For the maintenance of the corpora online, Spock comes with an integrated administration environment. The administration environment allows you to verify and add (import) transcription files, as well as edit the information on the speakers and the information regarding each sound file. The administration is web-based just like the front-end.

Contrary to the sound fragment files generated in the front UI, the admin environment has to deal with complete sound files. Given that these can easily get rather large (typically around 100Mb for a 30 minute file in 11kHz/16bits), these files are not uploaded via the web pages. They have to be uploaded separately via FTP or other transfer protocols supported by the server. After uploading the sound file(s), the transcription file has to be added to the corresponding files that have been uploaded via FTP, using an online form.

When importing an annotation file, the system parses the file to see which tiers or sound files are related to it, and asks the administrator to indicate for each of these which uploaded sound file it belongs to. It also gives you the option to exclude tiers from the import, as is for instance done with the tiers indicating background noise in the case of CORP-ORAL. For each speaker in the original annotation file, a record is added to the speaker database, which can afterwards be filled in with the relevant information about the speaker in question.

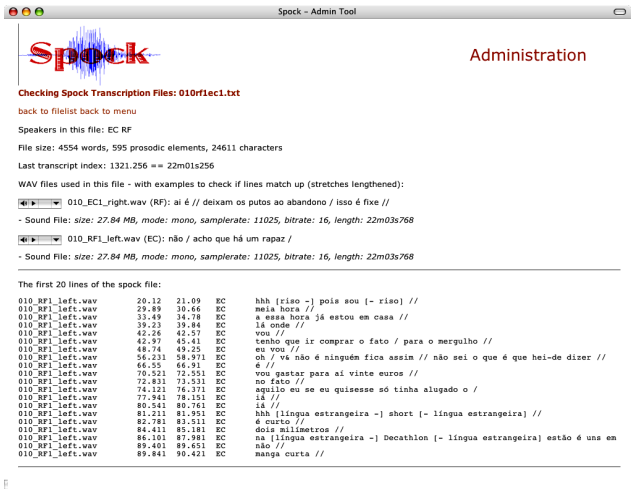

Figure 3: Screenshot of the Admin environment

Since the matching of the sound files and the annotation file is done manually, the system provides an info screen about each imported annotation file (see figure 3), to help verify whether the sound files and annotation file were correctly matched. For this, the system presents the total length of the sound file, as well as the last time index of each of the tiers. If the annotation is either longer than the sound file or much shorter, it displays a warning stating that the sound file is probably not the correct one for the transcript file. Furthermore, the system chooses a random line from each sound file linked to the transcription file, with a button to play its respective sound file. This should help to verify whether the alignment is correct after import, and whether for instance the bit rate of the sound file was not incorrectly parsed. And it displays the raw source of the first twenty lines of the generated transcription file, to verify for instance whether there were no problems with the character encoding.

Together with the data that are intended for verifying whether the import was successful, the information screen also presents some information of general interest about the sound file and the transcription file. For the transcription

file, it presents a list of all the speakers indicated in the file, the number of prosodic units, the number of words, and the number of characters. And for every sound file related to the annotation file, it presents the total file size, the mode (mono or stereo), the bitrate, the samplerate, and the total length in minutes and seconds.

4. Conclusion

We hope to have demonstrated in this article how Spock is a useful tool for making time-aligned corpora available to a larger audience. Although there are other tools available for distributing time-aligned corpora, the unique simple design of Spock should make it appealing both for general users and for corpus builders. The fact that the user does not need to install or download any special tool to access the spoken corpora made available with Spock means that it is much more immediately accessible. The lower threshold should attract users that otherwise would not go through the trouble of getting the aligned corpus to work. The easy and quick access to the data by means of simple queries means that it will be more attractive for less computer-savvy users, which should hopefully boost the use of spoken corpus data amongst a wider range of linguists, potentially leading to more corpus-based research. Even for the non-specialist, the web interface might be accessible enough to provide data to anyone interested in language use.

The fact that the package is very lightweight, and runs on any UNIX or LINUX based server, should mean that anyone who wants to make his time-aligned corpus available with Spock should be able to do so. The strains on the server both in terms of disk space and in terms of processing time are so low that any server should be able to handle it. And the import functions in the administration environment should be intuitive enough to allow anyone to convert their corpus into the format required by Spock.

Although Spock will never replace existing stand-alone transcription software, and also does not aim at attempting so, it should provide a useful addition for researchers developing time-aligned corpora, allowing them to bring their work to a larger audience than they otherwise might have reached.

4.1. Future Development

Although Spock in its current form presents all the features necessary for accessing time-aligned corpora in an orthography-based manner, we are trying to add more features to make it even more appealing as a corpus distribution tool. One of the things we are currently working on is trying to integrate a Part-of-Speech tagger into the administration system: although the system currently supports POS-tagged corpora, and provides a number of query options for them, the POS tags have to be manually crafted into the annotation file at the moment. We are trying to provide a way in which existing taggers such as TreeTagger or Brill can be used directly to automate this process.

Another area that we are currently exploring is to see how it would be possible to provide some simple acoustic analysis with the sound data. Our first attempt at the moment is to integrate some graphical F0 analysis to be displayed for each of the matching prosodic units in a query. This should make the system more readily usable for prosodic analysis of spoken corpora. For a lot of the basic acoustic analysis, the problem is not so much how to compute the data (since several open source tools are available for provided necessary data), but how to present the data online in an intuitive and user-friendly way.

5. References

M. Barlow. 2000. *MonoConc Pro (Concordance software)*. Athelstan.

C.E. Barras, Z. Geoffrois, Z. Wu, and M. Liberman. 2001. Transcriber: development and use of a tool for assisting speech corpora production. *Speech Communication*, 33:5 – 22.

P. Berck and A. Russell. 2006. Annex: a web-based framework for exploiting annotated media resources. In *Proceedings of LREC 2006, 5th International Conference on Language Resources and Evaluation*, pages 5 – 22.

P. Boersma. 2001. Praat, a system for doing phonetics by computer. *Glot*, 5:341 – 345.

H. Brugman and A. Russell. 2004. Annotating multimedia / multi-modal resources with elan. In *Proceedings of LREC 2004, 4th International Conference on Language Resources and Evaluation*.

J. Carletta, S. Evert, U. Heid, J. Kilgour, and J. Robertson H. Voormann. 2003. The nite xml toolkit: flexible annotation for multi-modal language data. *Behavior Research Methods, Instruments, and Computers*, 35:353 – 363.

T. Freitas and F. Santos. 2008. Corp-oral: Spontaneous speech corpus for european portuguese. In *Proceedings of LREC 2008*.

P. Martin. 2003. Winpitch corpus, a software tool for alignment and analysis of large corpora. In *Proceedings of the EMELD 2003*.

J.T. Milde and A. Thies, 2002. *The TASX Environment: Owners Manual*. Bielefeld University.

K. Sjölander and J. Beskow. 2000. Wavesurfer - an open source speech tool. In Yuan and Tang, editors, *Proceedings of ICSLP 2000, 6th International Conference on Spoken Language Processing*, pages 464 – 467.

N. Thieberger, 2007. *Audiamus Versions 1 and 2: A tool for building corpora of linked transcripts and digitised media*.

M. Vigário. 2003. *The Prosodic Word in European Portuguese*. Mouton.