Notes on Corpus Construction

R.J.J.H. van Son

Contents

Notes on Corpus Construction

R.J.J.H. van Son
AVL, Amsterdam & ACLC/IFA, University of Amsterdam
the Netherlands
R.J.J.H.vanSon@gmail.com

May 19, 2014

Abstract

Corpora of spoken language are both important for research, rare, and frequently difficult to use. These notes give some suggestions to ease the construction and distribution of corpora.

1 Introduction

During short term research projects there tends to come a moment where the question is raised whether and how to preserve the data gathered over the years. The present notes are intended to help researchers who want to preserve the data gathered in their projects and make these data available to other researchers. I want to suggest some tips that might help to improve the availability of original research data and ease the efforts needed to make them available. There are good books that go into detail about the planning and construction of language corpora [1, 2, 3]. These notes will be limited to practical tips on how to convert existing, small, data sets into a corpus.

The focus of these notes will be on mostly static data from small and short term research projects, e.g., by PhD students or Post-Docs, but the suggestions can easily be adapted to other situations. In such projects, the amount of data is limited and the time available to organize them is even more limited. The data should be organized in such a way that it can be stored indefinitely on a single departmental server. However, it should be easy to add more data to an existing corpus or to copy and move the whole collection from one location to another. For an example of how to construct a much larger corpus, please see the *Spoken Dutch Corpus* [4, 5] or the *DOBES* project [6].

These notes will discuss corpus construction based on a few example spoken language corpora that are available from the IFA Spoken Language Corpora [7, 8, 9, 10]. The topics that will be discussed are: Corpus structure, content and documentation, and distribution. These topics will be illustrated with a few examples

2 Corpus Structure

We assume a corpus that consists of files which are stored in directories. A corpus is a storage structure for (primary) data. As such it should help in bookkeeping of experimental data. The first question will be where to store the resulting corpus. The best location would be on a well maintained web server or the "cloud" (e.g., Dropbox [11]). In one stroke, such on-line storage would take care of many chores, from back-ups to distribution. But if such a solution is not available, you can start small, just a hard drive to work on is enough, and preferably, a second drive to back up to.

2.1 Recordings and media

The first decision to make when constructing a corpus is *What data should be stored in the corpus?* We can make a distinction between original media, e.g., texts, audio and video recordings, but also EEG recordings, and data connected to these media, e.g., annotations, subject responses or evaluations, and analysis results. Beside these data, there is also meta-data about the subjects involved and experimental circumstances. These can most easily be stored in a separate section of the corpus.

A decision that has to be made on this topic is the *unit of data* that should be stored. As everything is stored as files, each file should contain a *unit of data*. This can be a text, or a text fragment. It can be a sound recording, but also sections of a sound recording or even individual utterances or words. Storing several copies or parts of each item complicates the corpus considerably. It becomes time consuming to ensure that all instances of a certain item are correctly selected and updated when there is a correction or replacement. And it is almost inevitable that there will be corrections and replacements at some stage in the building and use of a corpus.

There is one exception to the advice to keep only single instances of every data item. When a corpus is constructed by processing original materials, e.g., edited video recordings or large texts, then it is advisable to keep archival copies of the original materials. However, these archival copies do not have to be part of the *real* corpus. This is especially important for video and audio recordings. It is very likely that such recordings will not be used in the format they were recorded in. Very often, video or audio compression has to be applied. The format conversions can lead to artifacts or loss of information [12]. Accessing the original data can then be necessary to be able to decide how to interpret ambiguous or suspicious observations.

2.2 Directories

The corpus will be organized in directories. Files of a single type that belong together will be stored in a single directory. This means, that in a video dialog corpus, all video recordings can be stored in a single directory. The extracted audio files will be stored in a different directory, as will be the transliterations of the speech and the annotations of, e.g., gaze direction. In more complex corpora, it could be better to group files in sub-, or sub-sub-directories. So, a corpus of pathological speech build from a group of separate studies could have separate sub-directories for different pathologies, and sub-sub-directories of the different tasks which were recorded, and sub-sub-subdirectories for the different studies from which the recording were taken.

The easiest way to organize such a complex corpus would be to make mirror directory trees for the different types of recordings (media), and transliterations and annotations. So, if video recording X.avi would be stored in directory *Video/A/B/C/D/X.avi*, then the associated audio file X.wav should be stored in *Audio/A/B/C/D/X.wav*, the EEG in *EEG/A/B/C/D/X.bdf*, the transliteration in *Translit/A/B/C/D/X.txt*, and the annotations in *Annotations/A/B/C/D/X.TextGrid*. Note that each directory contains files of uniform type. This greatly simplifies analysis, versioning, and backup. This scheme works best when files referring to the same recording file share the same name, e.g., like the *X* in the example.

2.3 File names

File names are tricky. It is best to ensure that every filename in the corpus is unique. If not, data will get lost if a file accidentally ends up in the wrong directory. When working with existing non-unique filenames in a more complicated corpus, there is an easy way to make them unique: Prepend the directory path in front of the filename. For example, if there are many files with duplicate names in different subdirectories, e.g., *Audio/A/B/C/D1/X.wav* and *Audio/A/B/C/D2/X.wav*, converting the filenames form *X.wav* to *A_B_C_D1_X.wav* and *A_B_C_D2_X.wav* would suffice to make every filename unique. Such a change can easily be scripted and automated in, e.g., *Praat* [13]. Note that using spaces in file names can seriously complicate automatic processing. It is best to use a character like '_' instead of a space.

It is also very convenient when the filename transparently indicates where it belongs and what it contains. For the example of a two camera recording in a dialogue video corpus, file names could start with *DVA[num][FM][age][A-Z]* for recordings of camera A and *DVB[num][FM][age][A-Z]* for the recordings of camera B. The *number* would be the number of the dialogue, *[FM]* would indicate Female or Male speaker, the age would give the age in years of the speaker, and the letter *[A-Z]* would indicate the speaker in view. So, any file whose name starts with *DVA14M62W* would be from camera *A*, dialogue *14*, and male, 62 year old speaker *W*. The corresponding partner recording in this would be *DVB14M72X*. The file type tells us what is stored in the file. *DVB14M72X*.dv for the uncompressed video, *DVB14M72X*.avi for the compressed video, *DVB14M72X*.wav for the audio. *DVB14M72X*.TextGrid for the annotations, etc. [8, 9]. Note that the above practice of prepending directory names in front of the filename to make the filename unique also makes it transparent where the file belongs in the corpus.

It is common to annotate or extract parts of the items in a language corpus, e.g., sentences and words from texts, or utterances and words from spoken language recordings. When such parts have to be named, it is very useful to prepend the item name with the (unique) name of the originating file. For example, finding

an utterance recording with the name *DVA14M62W1J* (eg, turn *1*, sentence *J* of recording *DVA14M62W*), it is easy to find the relevant (meta-) information in the corpus. This transparency is also useful for checking whether selections and (meta-) data are correct. Some quite elaborate item naming schemes can be found in [7, 8, 9].

3 Content and documentation

The contents of a corpus can be divided into three categories:

- *Primary data*
  Original recordings and observations
- *Meta-data and documentation*
  Information on Primary data
- *Derived content*
  Everything that can be reproduced from the other two

The first two, *primary* and *meta-data* are the real content of the corpus. The third category, *derived content*, is only stored because of convenience. The fact that *derived content* can be reproduced does not mean that it should be reproduced every time. Only that when *primary* and *meta-data* are changed or adapted, the dependent *derived content* can be regenerated. In storage and back-up procedures, *primary* and *meta-data* should be handled with extra care. *Derived content* often does not have to be backed up at all.

It must be emphasized that derived content should preferably be generated by automated methods, i.e., scripts. Manual labor is very difficult to reproduce. If an original recording is split up by hand into smaller parts, e.g., utterances, it will be very time consuming to generate a new set after even a minor change. However, if the original segmentation had been stored in a TextGrid file, then it would take only a few changes in the TextGrid annotations and running a script to regenerate the changed set. Moreover, anyone can check whether the original segmentation was indeed correct.

This can be generalized to other aspects of the corpus. Whenever possible, construction and maintenance of a corpus should be automated. Preferably with scripts or other documented means. These automated procedures should be organized in such a way that users of the corpus can maintain or reconstruct the corpus using as little external (insider) knowledge as possible.

3.1 Participants and procedures

In a language corpus, it is important to store all relevant data of the speakers, authors, and other subjects that participated in constructing the corpus. Up front, it is often difficult to decide what is relevant and what is not. So there might be an incentive to include as much information as possible. However, many pieces of information about subjects are privacy sensitive and should not be distributed or even included in the corpus. Such privacy sensitive material should be handled with extra care. Some personal data is considered so sensitive in some jurisdictions that it is illegal to collect and store them. Here we can give only some rules of thumb. Please, inform yourself about the laws in your jurisdiction.

Sensitive information should stay "in house" and not be distributed without a signed informed consent from the subjects. *Highly sensitive* data should only be stored in a secure environment. An easy way to prevent mishaps is to encode all subject names at the earliest possible moment and store (highly) sensitive data, like contact information, off-line or printed on paper, in a locked closet. At a later stage, relevant information can be compiled from the offlin storrage and anonymized for the corpus. There are special guidelines for working with data from children and medical records. The short version is that you should not share such data and keep everything behind locks or in a secure environment. The long version is that if you work with such data, you might want to re-read the relevant guidelines.

Every study will have its own requirements for data about the subjects and language. Often some aspects of language use and the subjects are not relevant for the original research. However, other users of the data might need such personal data. So it often pays off to record them anyway. Some types of data are almost always relevant. Data of subjects and circumstances that are generally relevant to construction and use of spoken language corpora can be listed as:

- Contact information of the subjects
  This is highly sensitive and should not be shared
- Age in years
  Date of birth is sensitive
- Sex/gender
- Language variant used, native language, other languages
  Place of origin, e.g., postal code (this can be sensitive if too specific)
- Hearing and speaking problems (mostly, the absence thereof)
  When relevant, the nature of any pathologies (highly sensitive)
- Recording: Date and Location of the recording and the Name of the person doing the recording
- Equipment and recorder settings

Some data, like the language or technical details of recordings and procedures, tend to be fixed in which case they can be stored in the general documentation.

It is very important to have all the subjects that participate in the creation of the corpus sign the relevant documents (see section 4). Most importantly are copyright forms that transfers all copyrights to the corpus maintainers and informed consents for speakers and experimental subjects. Make sure that these forms and declarations explicitly include a reference to the distribution of the materials [7, 8, 9].

3.2 File formats

In general, you should do as little conversions between file formats as possible. But if the aim of the corpus is to make the data available to outsiders, then it makes sense to chose file formats that are in common use. In general, plain text and all formats read by *Praat* [13] should be fine. Some applications, like office or specialist video software, produce files that are difficult to use without the exact same software or even computer platform. In such cases, it is best to add copies in generally readable format. Note that, at the time of writing, video codecs are a complete mess [14].

If possible, for non-generally readable file formats, ensure that copies using the the following formats are also available in the corpus:

- Word processor, e.g., *MS Word*: Plain text and PDF
- Spreadsheet and database, e.g., *MS Excel*, *MS Access*: Export as Tab Separated Values (.tsv) or Comma Separated Values (.csv)
- Audio: Uncompressed WAV files, else anything written by *Praat* [13] is good
- Pictures: PNG or JPEG files
- Video: Compress as little as possible, original DV or MJPEG would be nice, but at least use something that can easily be played with *VLC* [15] or *ELAN* [16] (note, *ELAN* video support is platform dependent)
- Annotations: *Praat* [13] TextGrid files or *ELAN* [16] EAF annotations

3.3 Annotations

Annotations are interpreted here as texts that are (time) aligned to the recordings [17]. These notes will discuss annotation files like those used in *Praat* [13] TextGrids and *ELAN* [16] EAF. An annotation generally has a start and end time, a *Tier*, and a text. Other set ups are possible, c.f., [17, 4, 5]. In corpus construction and maintenance, annotations have two independent functions. The first is to segment the recordings so relevant fragments, e.g., utterances, turns, or words, can be identified and retrieved. The second is to add additional information to the recordings. In general, annotations include a transliteration of the speech as text and a segmentation in utterances (or Inter-Pausal-Units), turns, and sometimes phrases or words. Other annotations used are (in random order): Gaze direction, word stress, backchannel utterances, disfluencies, gestures, or emotions. For adding any information related to fragments of the speech or language, standard

annotation files should be used wherever possible. These should be stored as text files (*short text file* in *Praat* [13]).

It is most efficient to use a segmentation stored in annotation files to split up recordings. It is fairly straightforward to code a *Praat* script that takes a TextGrid and copies out selected intervals for, e.g., listening experiments. This annotation file is then instant documentation of the segmentation and selection. The segmentation can also be easily adapted and recreated if needed.

3.4 Metadata and CMDI

Corpus data have only little value without information about the speakers, authors, task, and recording circumstances. This type of data is called *metadata*. In general, each item in a corpus has its own metadata. If metadata is available, it should be stored in its own, parallel directory tree, just like the annotations. The top level can be called *Info* or *Metadata*. Please consult the relevant standards documents when using browsable metadata hierarchies, [18, 19].

There is extensive literature about the use of metadata in corpus construction, eg, [20, 21, 22, 23, 24]. There are several international metadata standards for language corpora. Early ones are *TEI* [25] for text and *IMDI* [18] for multi media data. The *IMDI* standard has been adapted inside the Clarin project [26] to the *CMDI* standard [19, 27]. The advise is to use *CMDI* to code browsable metadata.

How to compile metadata is outside the scope of these notes. Readers can consult the Clarin user guide by Wittenburg and van Uytvanck [20] for more information as well as the relevant corpus handbooks [1, 2, 3]. It is important to remember the recomendations from [20]:

- Always start as early as possible to collect and create metadata. Otherwise chances are high that information is lost or that fixing the incomplete metadata records afterwards will be very costly.
- Try to achieve a high but reasonable level of granularity.
- Try to reuse as much as possible. It should save you work and will enhance the interoperability.
- Be aware that there are conversion methods in place for the most widely used formats there is no need to reinvent the wheel.

3.5 Scripts, programs, and applications

As much as possible the construction of the corpus and the analysis of the data should be automatized. This even holds for the statistical analysis and the production of figures for publication. Automatization is mostly done through the use of *scripts* or other programs. Experience shows that quite often, such scripts have to be consulted later to (re-)verify the validity of data and procedures, adapt the corpus or analysis to new requirements, and to reuse them for new tasks. Such scripts should be an integral part of the corpus. They will be necessary, not only to manage or rebuild the corpus, but also to understand the corpus and the analysis of the data. If special purpose programs have been used, it should be considered whether the version used in building or analyzing the corpus can be included with the corpus.

By adding the scripts in fixed directories, either in a separate *Scripts* tree of sub-directories, or as part of the *Documentation* sub-directory, it will be possible to use relative file paths. Relative file paths are necessary to allow the corpus to be moved from one computer to another. If at all possible, relative file paths should be used in scripts. See the *Praat* manual for examples of the use of relative file paths in scripts [28].

3.6 Documentation

Any information about the corpus that is not stored with the annotations or metadata is documentation. The aim of the documentation is to help users to understand the contents of the corpus, maintainers to extend and correct the corpus, and others to recreate a similar corpus. The documentation is also an archive for what has been done. This will be needed when data from the corpus is used for a publication. So the documentation should at least contain all information that is required for publishing a scientific paper about the contents of the corpus. The documentation section is also a good place to store information about the corpus itself, e.g., contact and license information, scripts and other special purpose software, errata, literature references, manuals. It is often convenient to store compilations of global metadata in the Documentations section, like speaker data and recording lists.

In a corpus, just create a sub-directory called *Documentation* and store any useful files in there. This directory can contain sub-directories as is convenient. Obvious content of a *Documentation* sub-directory are:

- Contact information, copyright and license documents of the corpus
- A changes and errata list
- A diagram of the corpus structure
- Anonymized lists of subjects: Speakers, listeners, and other experimental subjects
- Relevant parts of anonymized information about the subjects
- A list of recordings, e.g., with dates and the names/codes of those involved
- All technical details of the recordings or experiments, preferably with photographs
- Copies of any documents and forms given to the subjects, e.g., copyright forms and informed consent declarations (but *not* copies of the signed documents)
- Copies of any documents and manuals used during the construction, recording, and annotation of the corpus
- All scripts and special purpose programs used in the construction and analysis of the corpus (when not stored in a separate subdirectory)
- All publications based on the corpus
- A bibliography of the relevant literature, e.g., as a BibTeX reference file

3.7 Backup

The advised mind-set is to imagine, every day, that the building where your data reside burns down to the ground. Then think of how you would want to continue with your work. On regular moments, you should try out whether you really *can* continue your work after the building has burned down to the ground.

3.8 *Advanced topic*: Version control and repositories

A spoken language corpus generally consists of static language and speech data and a monotonically increasing amount of annotations, scripts, and other textual materials. The static language data can be backed up once and then the backup has to be updated only infrequently. However, it is almost always good to keep track of changes in the textual materials in a fine grained time scale. Errors happen and often changes will have to be undone. For this, versioning systems should be used [29]. In short, a version control system will store "snapshots" of your texts. It allows you to roll-back your system to any point in its history and even to pick and "undo" individual changes made to the system. The most important systems allow to merge changes made by different maintainers at different times. Note that version control systems are mainly useful for textual data, e.g., annotations and documentation. Most systems cannot well handle binary data, like pictures and word processor files.

A version control system will store the history of a project in a *repository*. Such a repository can easily be made available from a website or project server. Such repositories are excellent means to distribute and update corpora. A popular choice of version control system is *Git* [29]. See the slideshow at [**?**] for a short introduction. Explaining the use of version control systems is outside the scope of these notes. But there are excellent tutorials and manuals for most popular system, see the links and references in [29].

4 Distribution

The aim of a language corpus is giving other people access to language data. While designing and constructing a corpus, the ways the corpus will be accessed have to be taken into account. The corpora discussed here are small enough to distribute in full. Therefore, there is no pressing need to consider elaborate online search and selection solutions. For practical reasons, subsets of the corpus can be made "pre-canned" for download.

Otherwise, it is simplest to allow wholesale copying of the corpus to prospective users. The simplest technical solution is to install a web server, e.g., *Apache*, and point it to an index file in the top directory of the corpus. The technological details of this are beyond the scope of these notes, but you can look at the *Apache* documentation for more information [30].

Beyond the file format compatibility issues and useful documentation, there are a few legal matters that have to be dealt with before a corpus can be distributed.

4.1 Copyrights

In general, everything spoken or written will fall under copyright protection. In a language corpus, the language that are the primary data in the corpus are almost always protected by copyright law. When spoken language is involved, the speakers have a comparable right as "performers". So have the editors of the material. All the written materials and documentation will be protected under copyright law. This list could be extended ad infinitum. Under copyright law, anyone who wants to copy or distribute protected works. a corpus, needs written permission of the "owners" of the copyright. That would mean all those involved in constructing the corpus, which is impractical to say the least.

The solution is to ask everyone who participates to sign a copyright transfer form. In this form, the participant transfers all copyrights to the "owner" of the corpus, who will then be the sole owner of copyright. This new owner is some legal entity, that will manage and distribute the corpus. In many cases this entity will be some part of the university or research institute. However, it can be the creator of the corpus, or some non-profit organization, e.g., Nederlandse Taalunie. Just remember that this new owner will be the legal owner of the corpus. This entity will decide what will happen with the corpus. So some care might be taken to chose a suitable entity to transfer the copyrights to and to make good, binding agreements about the future of the corpus.

The question on who should all sign a copyright transfer form is not easy to answer. It is best to be "inclusive" and just ask every person who touches the data to sign a copyright form. If in doubt, ask for a signature. This policy works best when people give their signature *before* they start with their contribution. This includes all subjects who speak. However, there could be problems when spontaneous, or unscripted, speech is recorded. Then the speaker would have signed a copyright transfer before she or he knew what was said. In such cases it is best to confirm the signature again after the recording. That is, when recording unscripted language use, ask the speakers to sign the forms a second time after the recording. Sometimes it is prudent to give the speakers a copy of the recording and offer them the option to retract their permission or consent. For an extended discussion of this topic, see [8, 9].

Drawing up a copyright transfer form should be done by a specialist in copyright law. In most cases, this is too much work (and too expensive). It is then best to take a boilerplate form and adapt it to the needs of the corpus. It must be stressed that the transfer forms should make clear, upfront, to subjects how the recordings and the personal data might be used. In practice, this means that the different options, eg, publishing recordings and meta data on the internet, have to be written explicitly into the copyright transfer forms. A good guide seems to be that corpus creators are specific about the intended uses whenever possible. At the same time, an effort should be made to be inclusive and prepare for potential, future, uses by yourself and others. All the "legal" information has to be made available also in layman's terms in an informed consent declaration (see below). Obviously, subjects should have ample opportunity to ask questions about the procedures and use of the recordings. An example copyright transfer form can be found at the IFA Dialog Video corpus [10].

4.2 Informed consent

Having experimental subjects, or speakers, sign legal documents does not imply that they fully understand the effects their participation can have on their lives. However, it is paramount that all subjects fully understand the potential consequences of participating. To ensure that every subject has understood and accepted the (potential) consequences of their participation, they should sign an *informed consent* document, e.g., [31]. This document should explain in plain and easy to understand language what will be expected from the participants, and what the consequences can be of their participation. This informed consent document should also state what will happen with their contributions. For example, if spontaneous dialogue is recorded and will be published on-line, it should be ensured that the speakers know about this. If such a publication on-line might

possibly affect their future life or career prospects, this should also be made clear to the subjects (and ways should be found to prevent or remediate such outcomes).

Journals and conference publishers generally require informed consents from all subjects whose data are used in a publication. Special consent is required when pictures or video clips of subjects are used in a publication or presentation. In practice, *both* signed copyright transfer forms *and* informed consent documents are required before any data in a corpus can be used. And it cannot be stressed enough that both documents should contain clauses about *all* intended and possible future uses of the data.

4.3 Privacy

It is generally accepted that researchers have a *duty of confidentiality* with respect to informants and experimental subjects and should respect their privacy, e.g., [31]. Only in exceptional cases will the names or other identifying information of experimental subjects be revealed. In many cases it is even prohibited by law to reveal the identities of subjects, e.g., of minors or patients. When co-authors or colleagues participate in experiments, it is common to use their initials in publications. However, names of external subjects should always be securely coded and initials are not considered secure. For an individual paper, enumerating subjects, e.g., S_1-S_i, often suffices. In a corpus, this can become unwieldy, especially in more complicated corpora containing multiple contribution from individual subjects, sometimes in different roles. Unless ethical rules require total anonymization, each subject should get an individual, fixed code that is valid for the complete corpus. It is obvious that the real names and contact information should never be stored with the corpus.

If at all possible, subjects should get a (random) code at enrollment. Trying to fix the internal codes after recordings of experiments are completed is fraught with problems and frequently leads to persistent errors. The easiest system for subject coding is simply giving numbers or letters in order of enrollment. This can be $[S1\dots]$, or $[A, \cdots, ZZ]$. It does not matter for such a procedure that some subjects will drop out. In special cases, it can be necessary to adopt more involved protocols. Needless to say that the decoding lists that link subject codes to identities and contact information should be stored and backed up securely.

An example of a special case was the use of patients enrolled in long term follow up research. Many patients participated in several studies while it was not always clear which patients had already participated before. Unique subject codes were constructed by encrypting the hospital patient ID and date of birth (using a password). This ensured that every patient contribution for every study would be labelled with the same subject code even when the previous contributions were not known. The encryption procedure was designed to make it impossible to extract personal information from the codes. Cryptographic protocols are very fragile and can easily fail. Caution must be exercised when using them. A demonstration project for constructing anonymous ID's can be tested here [32].
A few suggestions for handling subject privacy:

- Keep all personal and contact information off-site, make sure there is a printed paper back up
- Remember that date-of-birth and (Dutch) postal codes are likely to be sensitive information
- Assign each subject a unique anonymous code at enrollment
- Subject codes should be sequential, random, or if that is not possible, cryptographically secure
- Use subject codes for *all* references to subjects, internal and external
- Do not publish recognizable audio, pictures, or video without the explicit consent of the subjects
- If sensitive information *has* to be shared with outsiders, require a signed *promise of confidentiality* [31]

4.4 Licenses and moratoria

As discussed in section 4.1, the owner of the copyrights to the corpus must give written permission for use of any part of the corpus. This can be on a case-by-case basis, which is impractical, or more efficiently by way of a copyright license. A copyright license is a written permission to copy and distribute work under copyright, i.c., a corpus. A copyright license determines how a corpus can be used and by whom, and what can be done with the results. These notes will only discuss the *Open Data* case [33], where a corpus is shared on liberal terms.

There are two families of copyright licenses relevant for *Open Data* compatible corpora, *Free and Open Source* licenses for code and software [34, 35], and *Creative Commons* licenses [36] for all other materials. When choosing a license, it is strongly advice to adopt an existing license. In practice, adopting a newly written license has only downsides. Lists of popular licenses can be found at:

- *Creative Commons* [36]: `http://creativecommons.org/licenses/`
- *Free and Open Source* [35]: `http://opensource.org/licenses`

It is often impractical for a researcher to wait until her project is completed before adding her data to an *Open Data* corpus. For logistic reasons, it might be even preferable to store all primary data directly in an existing corpus. In such cases, the researchers would not allow distribution of their data before they have finished their primary analysis and publication. This is handled by a *moratorium* on the data. When the data are added to the corpus, the "license" specifies an agreed date, after which the data will be available to the "public". Alternatively, the data will be made available after the official publication of a certain paper. When constructing a corpus, the possible inclusion of such moratoria should be considered.

5 Examples

5.1 Example: Simplified *Proper Speaker Turn Switches*

5.2 Example: Gaze direction

5.3 Example: Functional annotation

6 Conclusions

7 Acknowledgements

This work is made possible by an unrestricted research grant from Atos Medical (Horby, Sweden).

References

[1] M. Wynne, *Developing linguistic corpora: a guide to good practice*. Oxbow Books, 2005. `http://www.ahds.ac.uk/guides/linguistic-corpora/index.htm`.

[2] D. Gibbon, R. K. Moore, and R. Winski, *Handbook of standards and resources for spoken language systems*. Walter de Gruyter, 1997.

[3] D. Gibbon, I. Mertins, and R. Moore, *Handbook of multimodal and spoken dialogue systems: resources, terminology, and product evaluation*. Springer, 2000.

[4] N. Oostdijk, "The spoken dutch corpus. overview and first evaluation.," in *LREC*, 2000.

[5] N. Oostdijk, "The design of the spoken dutch corpus," *Language and Computers*, vol. 36, no. 1, pp. 105–112, 2001.

[6] P. Wittenburg, U. Mosel, and A. Dwyer, "Methods of language documentation in the dobes project.," in *Proceedings of LREC 2002*, 2002.

[7] R. van Son, D. Binnenpoorte, H. van den Heuvel, and L. Pols, "The IFA corpus: a phonemically segmented Dutch Open Source speech database," in *Proceedings of EUROSPEECH 2001 Aalborg*, pp. 2051–2054, 2001.

[8] R. van Son, W. Wesseling, E. Sanders, and H. van den Heuvel, "The IFADV corpus: a free dialog video corpus," in *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)* (N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, and D. Tapias, eds.), (Marrakech, Morocco), European Language Resources Association (ELRA), may 2008. `http://www.lrec-conf.org/proceedings/lrec2008/`.

[9] R. van Son, W. Wesseling, E. Sanders, and H. van den Heuvel, "Promoting free Dialog Video Corpora: The IFADV Corpus Example," in *Multimodal Corpora* (M. Kipp, J.-C. Martin, P. Paggio, and D. Heylen, eds.), vol. 5509 of *Lecture Notes in Computer Science*, pp. 18–37, Springer Berlin Heidelberg, 2009. http://dx.doi.org/10.1007/978-3-642-04793-0_2.

[10] Institute of Phonetic Sciences, Amsterdam, "IFA Spoken Language Corpora." http://www.fon.hum.uva.nl/IFA-SpokenLanguageCorpora/, 2013.

[11] Dropbox, "Dropbox." https://www.dropbox.com/, 2013.

[12] R. J. J. H. Van Son, "A study of pitch, formant, and spectral estimation errors introduced by three lossy speech compression algorithms," *Acta acustica united with acustica*, vol. 91, no. 4, pp. 771–778, 2005.

[13] P. Boersma and D. Weenink, "Praat: doing phonetics by computer." http://www.praat.org/, 1992–2008.

[14] S. Phipps, "Video codecs: The ugly business behind pretty pictures." http://www.infoworld.com/d/open-source-software/video-codecs-the-ugly-business-behind-pretty-pictures-214525, 2013.

[15] VideoLAN, "VLC media player." http://www.videolan.org/, 2013.

[16] ELAN, "ELAN is a professional tool for the creation of complex annotations on video and audio resources." http://www.lat-mpi.eu/tools/elan/, 2002–20013.

[17] S. Bird and M. Liberman, "A formal framework for linguistic annotation," *Speech communication*, vol. 33, no. 1, pp. 23–60, 2001.

[18] IMDI, "ISLE Meta Data Initiative." http://www.mpi.nl/IMDI/, 1999–2007.

[19] CLARIN ERIC, "Component Metadata CLARIN ERIC." http://www.clarin.eu/content/component-metadata, 2013.

[20] P. Wittenburg and D. van Uytvanck, "Chapter 2. Metadata." http://media.dwds.de/clarin/userguide/text/metadata.xhtml, 2013.

[21] L. Burnard, "Metadata for corpus work," in *Developing linguistic corpora: a guide to good practice* (M. Wynne, ed.), Oxbow Books, 2005. http://www.ahds.ac.uk/guides/linguistic-corpora/index.htm.

[22] E. Duval, W. Hodgins, S. Sutton, and S. L. Weibel, "Metadata principles and practicalities," *D-lib Magazine*, vol. 8, no. 4, p. 16, 2002. http://www.dlib.org/dlib/april02/weibel/04weibel.html.

[23] B. Hughes, "Metadata quality evaluation: Experience from the open language archives community," in *Digital Libraries: International Collaboration and Cross-Fertilization*, pp. 320–329, Springer, 2005.

[24] D. Broeder, M. Kemps-Snijders, D. Van Uytvanck, M. Windhouwer, P. Withers, P. Wittenburg, and C. Zinn, "A data category registry-and component-based metadata framework.," in *Proceedings of LREC 2010*, 2010. http://www.windhouwer.nl/menzo/professional/papers/metaData.pdf.

[25] Wikipedia, "Text Encoding Initiative (TEI)." https://en.wikipedia.org/wiki/Text_Encoding_Initiative, 2013.

[26] Steven Krauwer, "CLARIN ERIC: Common Language Resources and Technology Infrastructure." http://www.clarin.eu/, 2013.

[27] P. Wittenburg and D. van Uytvanck, "Chapter 2. Metadata: The Component Metadata Initiative (CMDI)." http://media.dwds.de/clarin/userguide/text/metadata_CMDI.xhtml, 2013.

[28] P. Boersma, "Scripting 6.4. Files." http://www.fon.hum.uva.nl/praat/manual/Scripting_6_4__Files.html, 2013.

[29] J. Meloni, "A gentle introduction to version control." http://chronicle.com/blogs/profhacker/a-gentle-introduction-to-version-control/23064, 2010.

[30] Apache, "Apache HTTP Server Version 2.4 Documentation." http://httpd.apache.org/docs/2.4/, 2013.

[31] L. Corti, A. Day, and G. Backhouse, "Confidentiality and informed consent: Issues for consideration in the preservation of and provision of access to qualitative data archives," *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, vol. 1, no. 3, 2000. http://www.qualitative-research.net/index.php/fqs/article/view/1024.

[32] R. van Son, "Create an anonymous ID." http://www.fon.hum.uva.nl/rob/AnonymousID.html, 2013. A demonstration site.

[33] Wikipedia, "Open Science Data." https://en.wikipedia.org/wiki/Open_science_data, 2013.

[34] "Free Software Foundation." http://www.fsf.org/licensing/, 2013.

[35] "Open Source Initiative." http://opensource.org/, 2013.

[36] "Creative Commons." http://creativecommons.org/, 2013.