# DitAT: a flexible tool to support web-based dialogue annotation

Jeroen Geertzen
Dept. of Communication & Information Sciences
Tilburg University, The Netherlands
j.geertzen@uvt.nl

## 1   Introduction

The interest in studies using human-human and human-machine dialogue material and the different requirements for different studies have resulted in a rich availability of dialogue annotation tools to facilitate the work of annotators. Examples are DAT [1] and MUP [3]. In this paper, an annotator-friendly and flexible dialogue annotation tool is presented that uses a web-based environment to distribute dialogue material, to collect annotations, and to allow instant evaluation.

DitAT is a tool that allows to annotate transcripts or texts with (supportive) audio playback. It can be used on several platforms such as MS Windows and GNU/Linux and provides a simple and user-friendly graphical user interface. The tool allows for context-sensitive help for the annotation classes, allows commenting on utterance annotations and offers the possibility of audio playback of dialogue fragments if audio is available. For data input and output, it supports both tab-separated fields in plain text (for easy pre and post processing[1]) and XML formatted data (including stand-off markup). At the moment, representing overlapping speech over multiple tiers or layers is not supported. The most important discriminative features, however, are support for more complex properties of dialogue annotation schemes (such as the general purpose functions in DIT [2]) and the usability requirements for both annotation and evaluation that have been considered :

---

[1]Whereas XML formatted data is excellent for data exchange, the need of XML parsing and bigger sized data files make it less attractive for easy pre and post processing.

1. *Minimalistic interface.* The annotation interface should only have the absolute required controls and should be simple;

2. *Dynamic interface.* The annotation interface should be changed dynamically to represent the characteristics of the annotation tagset that has been loaded;

3. *Easy session setup.* Annotators should not have to use, copy, or manipulate the dialogue data files (transcripts, audio, annotations) prior and after annotating;

4. *Instant presentation* The annotations that are made for the same data by multiple annotators should instantly be presented for discussion purposes. Also statistics concerning inter-annotator agreement (e.g. $\kappa$-scores) and reported difficulties should be available during and directly after annotation;

5. *Remote use* Dialogue annotation should be possible remotely with just a working internet connection;

Requirement two is met by representing all tags that can be assigned to a specific class in drop-down list boxes, where possible hierarchical relations are expressed with a tree-like indentation. Requirements three, four, and five are met by a communication protocol that enables the tool to exchange data using the internet.

## 2    A closer look at DitAT

DitAT consists of two components: an annotation program (Figure 1) that runs on the workstation of the annotator and a CGI application that runs on a HTTP server. The functionality of the annotation program has been briefly described in the previous section. As illustrated in Figure 2, the annotation program is designed to communicate over the LAN or Internet with the CGI application in order to request, distribute, and collect dialogue material and to present annotator comparisons, statistics[2], and co-occurrence matrices to support (group) analysis of the annotations and tagset.

The dialogue materials such as transcripts and speech audio fragments are centrally stored and retrieved by the annotation program when necessary. At the end of a session, the annotation program submits the annotations to

---

[2]On inter-annotator agreement with e.g. $\alpha$ and $\kappa$ scores and percentage agreement, but also on timing (number of utterances per minute) and reported confidence scores.

the HTTP server. Via dynamic web-pages, the submitted annotations made by different annotators on the same material can be immediately compared. Also inter-annotator agreement between all annotator pairs and overall average statistics are presented and co-occurrence matrices to support analysis of annotation classes are generated.

# 3    Conclusion

In this paper, a web-based annotation tool was presented that was originally developed to support dialogue annotation sessions for reliability and scheme revision studies. Although it aims to support dialogue annotation, it can be used for any annotation task on serially ordered text fragments or speech fragments[3]. The tool has been used successfully in the annotation sessions reported in [2] and is still under development. Currently, the tool is being refined to visualize discontinuous turns in a better way and a portal is being developed to enable other researchers to run web-based annotation sessions.

More information on use of the dialogue annotation tool can be found at: `http://www.cosmion.net/jeroen/software/ditat/`.

# References

[1] James Allen and Mark Core. Draft of DAMSL: Dialog act markup in several layers. Unpublished manuscript, 1997.

[2] Jeroen Geertzen and Harry Bunt. Measuring annotator agreement in a complex hierarchical dialogue act annotation scheme. In *Proceedings of the 7th SIGdial Workshop on Discourse and dialogue*, pages 126–133, Sydney, Australia, July 2006.

[3] Michael Glass and Barbara Di Eugenio. Mup: the uic standoff markup tool. In *Proceedings of the 3rd SIGdial workshop on Discourse and dialogue*, pages 37–41, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

---

[3]However, annotation of prosodic information would allow more accurate audio playback facilities than can be offered.
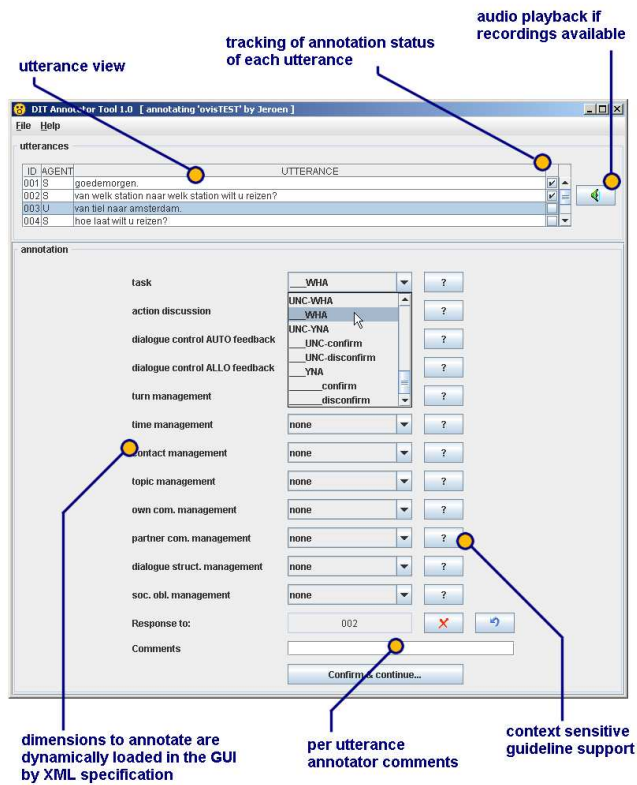
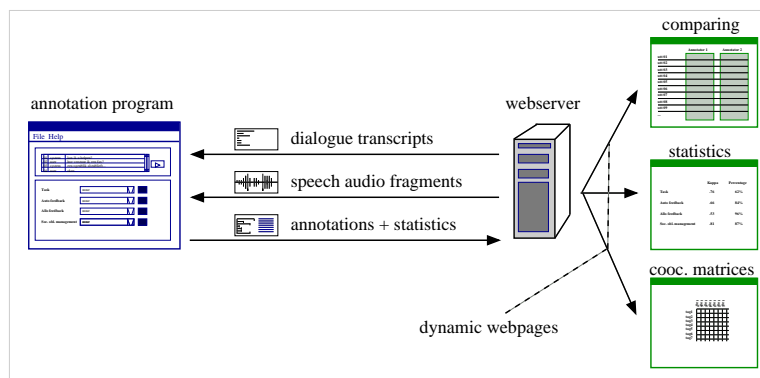Figure 1: A screen shot of the GUI of the dialogue annotation tool.



Figure 2: The exchange of information in DitAT.