

MTC: A GUI-based Utility for Analyzing Patterns in Dialog

Allison Penner
penneraa@bethelks.edu

CRA-W Distributed Mentoring Program Summer Research Project
Mentor: Professor Susan W. McRoy
Natural Language and Knowledge Representation Research Group
Electrical Engineering and Computer Science Department
University of Wisconsin-Milwaukee

1 Motivation and Introduction

Studies of human discourse involve data sets that must be transcribed and annotated in a tedious and labor-intensive fashion in order to perform the analysis. New, multimedia data sets, involving video and audio recordings as well as multiple levels of annotation, are being made available. However, as more complex phenomena are studied, the difficulty of annotation and analysis increases. Quek et al. (2002) have created multimedia software that simplifies the task of synchronizing and analyzing multi-modal data sets with overlapping levels of interpretation. The data prepared by their tool promises to be a rich source of new insights into various discourse phenomena, including information about speech as well as correlations to gaze (eye movement) and gesture (hand and body movement) in the context of a dialog. Multi-Tier Comparer (MTC) provides several tools to help in the quantitative analysis of annotated discourse data. It is meant as a utility specifically to look for patterns of argumentation in dialog situations where the two agents disagree, although it could be useful for any kind of data set that is divided up into time intervals and each interval given a label on one or more levels.

Annotating and analyzing a data set is a tedious task, and one that needs to be automated as much as possible if it is to be useful. Finding data about argumentation patterns is especially challenging because some sort of annotation must be used to codify each utterance in terms of its function in the argument as well as its semantic meaning. At the beginning of this kind of research, it is helpful for more than one person to annotate the same dialogs in order to establish inter-annotator reliability and come up with a well-accepted annotation. There is already tool called Praat (Boersma 2001) for the annotation of sound files which is in wide-spread use. Praat allows users to annotate a dialog on multiple levels, or tiers. Each tier can be extracted to an ASCII file using a Praat utility script. Each ASCII file contains a list of the intervals on that tier, including the start time and label of each interval. Files that have been extracted from Praat annotations are used as input to MTC. It provides a graphical comparison of multiple annotations of the same dialog in table format and the opportunity for the user to resolve the differences between the annotations. Given an annotated data set, MTC currently also provides tools to generate a list of label bigrams and their frequencies, as well as correlate events across annotation levels.

2 Functionality

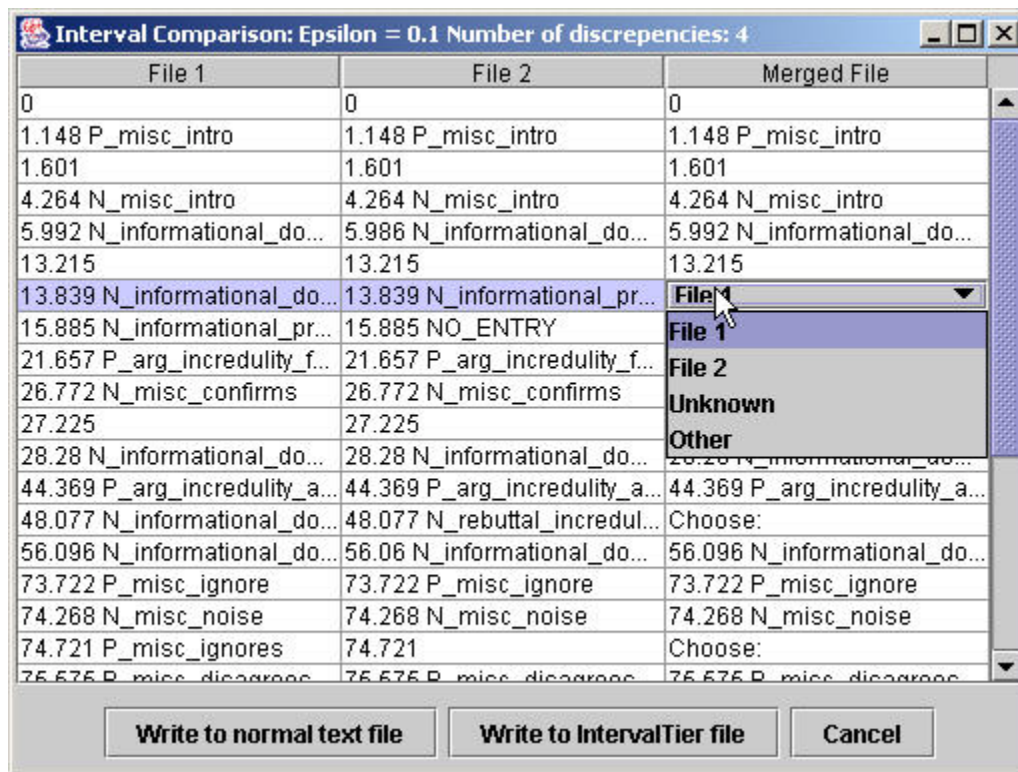
MTC has three main functions:

- Comparing multiple annotations of the same dialog
- Generating a list of bigrams and their frequencies across a data set
- Generating a list of co-occurring labels across annotation levels and their frequencies

2.1 Comparing tiers

MTC provides a tool that displays two or more files in table format, with one start-time per row and one file per column. The last column provides an opportunity to resolve differences between the two files and come up with one set of labels (fig. 1).

An added complexity to the problem was the fact that the start-times of two intervals may be slightly different although referring to the same utterance. This was taken into account by defining start-times to be equal within an epsilon value, which has a default value of 0.1 but can be set depending on the needs of the specific data set. The output can be written to two types of ASCII files: either in the same format as input files to MTC, or as a Praat-style short text file.



File 1	File 2	Merged File
0	0	0
1.148 P_misc_intro	1.148 P_misc_intro	1.148 P_misc_intro
1.601	1.601	1.601
4.264 N_misc_intro	4.264 N_misc_intro	4.264 N_misc_intro
5.992 N_informational_do...	5.986 N_informational_do...	5.992 N_informational_do...
13.215	13.215	13.215
13.839 N_informational_do...	13.839 N_informational_pr...	File 1
15.885 N_informational_pr...	15.885 NO_ENTRY	File 2
21.657 P_arg_incredulity_f...	21.657 P_arg_incredulity_f...	Unknown
26.772 N_misc_confirms	26.772 N_misc_confirms	Other
27.225	27.225	
28.28 N_informational_do...	28.28 N_informational_do...	
44.369 P_arg_incredulity_a...	44.369 P_arg_incredulity_a...	
48.077 N_informational_do...	48.077 N_rebuttal_incredul...	Choose:
56.096 N_informational_do...	56.06 N_informational_do...	
73.722 P_misc_ignore	73.722 P_misc_ignore	
74.268 N_misc_noise	74.268 N_misc_noise	
74.721 P_misc_ignores	74.721	Choose:
75.575 P_misc_disagree	75.575 P_misc_disagree	

Fig. 1: Graphical comparison of two annotations of a single dialog

2.2 Counting bigrams

A bigram is simply a pair of labels. By default, MTC generates a list of consecutive pairs of labels in the selected file(s), sorted by frequency. Sometimes it may also be useful to look at bigrams that are not necessarily consecutive but occur within a certain “window” of each other. MTC also allows the user to set an integer window size greater than or equal to two. The resulting list of bigrams can be written to an ASCII file.

2.3 Counting co-occurrences between tiers

In order to discover patterns in a dialog, it may be interesting to correlate events from one tier of labeling with the simultaneous label from another tier. For example, it may be interesting to count whether more speech disfluencies happen during questions than during other types of utterances. MTC provides a tool that generates a list of the events listed in one file and the label that was going at that time in another file. The pairs of labels are sorted by frequency if any pair occurs more than once in the pair of files. The list of label pairs can be written to an ASCII file.

3 Implementation notes

MTC is implemented in Java using the Java Swing package. This allows for extremely portable graphical interfaces. The input format also provides flexibility because ASCII input files can be generated by any program. MTC's object-oriented design will hopefully allow other researchers to easily incorporate additional analysis tools into the main program.

4 Conclusion

MTC is a GUI-based tool that is meant to aid in the analysis of multi-modal discourse data sets. It provides interfaces to compare multiple annotations of the same dialog, generate lists of bigrams and their frequencies across a data set, and generate frequency lists of pairs of labels that occur simultaneously across two levels of annotation. It will hopefully be a useful tool in examining the rich new multi-modal data sets that are being generated by researchers in the fields of psycholinguistics, natural language processing, and other related areas.

References

- Boersma, P.P.G. (2001): "Praat, a system for doing phonetics by computer". *Glott International* 5: 341-345. With a review by V. van Heuven.
- Quek, Francis K.H., Robert K. Bryll, Cemil Kirbas, Hasan Arslan, David McNeill (2002). "A Multimedia System for Temporally Situated Perceptual Psycholinguistic Analysis." *Multimedia Tools and Applications* 18(2): 91-114.