# TESTING THE ADEQUACY OF QUERY LANGUAGES AGAINST ANNOTATED SPOKEN DIALOG

Steve Cassidy[1]      Pauline Welby[2]      Julie McGory[2]      Mary Beckman[2]

[1]Speech Hearing and Language Research Centre, Macquarie University, Sydney
[2]Department of Linguistics, The Ohio State University, Ohio, USA

ABSTRACT: Large annotated collections of speech data are now common in spoken language research and a recent focus has been on the development of annotation standards and query languages for these annotations As part of this process it is important to evaluate the emerging proposals against a range of Linguistic annotation practices and in many different domains.

This paper presents an example of a richly annotated discourse segment which includes both DAMSL style discourse level annotation and ToBI intonational analysis. We describe how this annotation could be realised in either the Emu, MATE or Annotation Graph formalisms.

In order to evaluate the different query languages we take a small number of queries and attempt to express them in each query language. We are particularly interested in the naturalness of the query expression in each case. In some cases we find that queries cannot be expressed in the current language. We make a number of suggestions to guide the development of these query languages.

## INTRODUCTION

Large annotated collections of speech data are now common in spoken language research and a recent focus has been on the development of annotation standards and query languages for these annotations (Cassidy and Bird, 2000; Bird and Liberman, 2000; McKelvie et al., 2000). As part of this process it is important to evaluate the emerging proposals against a range of Linguistic annotation practices and in many different domains.

The Emu speech database system (Cassidy and Harrington, 1996; Cassidy and Harrington, 2000) has been in use for a number of years largely in small scale database projects relating to acoustic phonetic analysis of segments taken from short utterances and isolated words. While we believe that the annotation model provided by Emu is sufficiently rich for a wider range of applications, the annotation tools and query language associated with Emu are not adequate, for example, for the annotation of long multi-speaker dialogs. As part of the redesign and extension of the Emu system, we are evaluating the requirements of different annotation domains with respect to annotation tools, the data model and the query language. This paper presents the results of our analysis of a two speaker dialog which has been annotated for both dialog structure (DAMSL) and intonational features (ToBI). We are particularly interested here in how well this annotation fits into the data models proposed by two recent systems (MATE (McKelvie et al., 2000) and Annotation Graphs (Bird and Liberman, 2000)) and how well the query languages proposed for these systems handle sample queries in this domain.

While Emu emphasises an hierarchical view of the annotation, it can be shown (Cassidy and Harrington, 2000; Cassidy and Bird, 2000) that the data model is entirely equivalent to that in the annotation graph formalism. Hence our comments below on the AG model apply also to Emu.

## DIALOG ANNOTATION

### DAMSL

The annotation structure proposed in the DAMSL scheme (Allen and Core, 1997) divides a dialog into utterances units which correspond to a single speaker turn. Each utterance is tagged with a number of
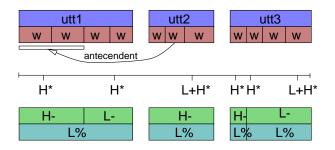
Figure 1: An example of the annotation structure applied in this study. Annotation elements are shown as extents in time but we assume an implied hierarchical relation between, for example Utterances and Words.

attributes which describe the role of the utterance in the dialog and which contain both forward and backward references to other utterances. The attributes we have included from DAMSL are: information level (info), forward looking function (flf) and backward looking function (blf). The values of these attributes are selected from a fixed set which is subdivided into groups; for example, the forward looking function can be a Statement (Assert, Reassert or Other-statement) or Committing Speaker Future Action (Offer or Commit) or one of three other groups of values. Importantly, these attributes may have more than one value since a single utterance may serve more than one function in a dialog.

The blf attribute describes a relation between the utterance and an antecedent earlier in the dialog. A value of, for example Accept is also associated with a reference to an earlier utterance that is being accepted.

Co-reference Annotation

We have included co-reference annotation at the word level to relate pronouns and their antecedents. In our example, pronouns are linked to their antecedent NPs occurring in an earlier utterance. This can be written using a tagged bracket notation as follows:

    S: [That number]-k is 1800 596 237.
    T: Okay, I got [it]-k.

ToBI

One of our motivations for investigating new tools for dialog analysis was to be able to combine dialog level annotation with the more detailed analysis provided by the ToBI system. ToBI analysis provides a word level segmentation of the dialog and adds tone level pitch event labels denoting both pitch accents and phrase tones. While ToBI has traditionally been realised as a flat annotation system the presence of intermediate and intonational boundary tones record an inherently hierarchical structure; it is natural to think of a ToBI analysis as describing one or more intonational phrases (L%, H%) dominating intermediate phrases (L-, H-) which dominate words.

Using this style of ToBI annotation in our example dialog means that there are now two intersecting hierarchies both of which dominate the word level transcription. This is illustrated in figure 1 which shows intermediate and intonational phrases as extents dominating word level tokens (boundary tones are omitted for clarity in this figure). ToBI also provides the tone, break and misc levels which bear no direct relation to the word level except, perhaps, that the tone level pitch accent labels might be associated with words. Since the utterance unit is a unit of discourse structure there is no reason to expect that an utterance unit will correspond to a single intonational phrase.

ANNOTATION SYSTEMS

Two annotation models will be considered to assess their ability to represent this dialog annotation and the expressive power of their associated query language.

MATE

The data model of MATE (McKelvie et al., 2000) is closely tied to that of XML but is generalised to allow for multiple intersecting hierarchies. The two intersecting hierarchies in our annotation are readily accommodated into MATE and would be stored in two separate XML files, one for the DAMSL annotation and another for ToBI. Only one of these files would contain word level labels, the other would contain hypertext references to the other file; these references are resolved by MATE when the annotations are read.

Relations between tokens are defined by attribute values using the XPointer notation. XPointer, which is a W3C standard associated with XML, allows any token in the annotation structure to be addressed unambiguously. It is not clear however whether named relations are supported, since the only examples of relations described in the published work on MATE are parent-child relations. Clearly, any attribute could take an Xpointer value to refer to another token but the only relation support provided in the query language is for parent-child relations.

Annotation Graphs

The annotation graph (AG) formalism was introduced by Bird and Liberman (Bird and Liberman, 2000) as a generalised representation scheme for Linguistic annotations. An AG consists of a set of nodes, representing time points, and a set of labelled arcs, representing tokens. Domination is represented implicitly by structural inclusion of one arc within another. Arc labels are stored as a simple attribute value list with one distinguished attribute to define the label type (e.g. Word or Utterance).

The annotation graph formalism does not define an explicit relation between tokens but suggests implementing relations using equivalence class labels. For example, a pronoun and its antecedent might share a common attribute value antecedent123 to indicate their relationship.

SUITABILITY OF THE DATA MODELS

The annotation structure we have presented fits well into each of the annotation formalisms under review. The only issues that arise in each case are the use of multiple values for attributes and the implementation of relations between utterances and between words.

In this discussion we will use the term token to refer to a labelled segment of speech and relation to refer to an association between tokens.

The dialog annotation outlined above requires that attributes be allowed to take on multiple values whereas in all three data models outlined here only single values are defined for attributes. A simple solution to this problem is to adopt the convention of recording multiple values as a space or comma delimited list. The weakness of this solution becomes apparent when queries to the database are considered since the query system must provide support for checking any of the stored values in a query condition.

All models support some kind of non-hierarchy relation between tokens, although in the MATE system, all relations seem to be treated as parent-child relations. The interesting feature of the DAMSL dialog annotation is that the values of the blf (backward looking function) attribute include a reference to earlier utterances. For example:

```
Utt:    3
Talker: T
Text:   I need to um book a hotel room.
info:   task
blf:    answer, accept (utt 1)
flf:    assert, action-directive
```

this utterance is both an answer to an earlier utterance and an acceptance of the offer of help. One could also imagine a case when one utterance was related to two earlier utterances; for example, answering one utterance and accepting another. Hence the data model needs to be able to store not only multiple values in attributes but multiple relational values.

If a relation is implemented as a reference to another token, either as an equivalence class, a token identifier or an Xpointer value, then there also needs to be a way of identifying the type of that relation or associating the relation with a particular attribute value.

```
($a = word)
($p = word)
($u1 = utterance)
($u2 = utterance);

($p.pos ~ "pronoun") &
($p.antecedent ~ $a.referents ) &
($a ^ $u1)  & } $p ^ $u2) &
($u1.id + 3 $>$ $u2.id)
```

```
($u = utterance)
($w = word)
($t = tone)
($i = intonational);

($u.flf ~ "info-request") &
($u ^ $w) & ($i ^ $w) &
($i @ $t)
```

Figure 2: The example queries expressed in the MATE query language Q4M.


DATABASE QUERIES

One of the primary reasons for completing a corpus annotation project is to be able to query the corpus to select exemplars according to their place within the annotation. Each of the formalisms described above provides a query language that can be used to locate tokens of interest within the annotation and produce a report including information about these tokens. The Emu query language has been in use for a number of years but it is now evident that it does not have sufficient expressive power for general purpose applications. Part of the motivation for this study is to evaluate the features required by a query language in order that a new language can be developed for the Emu system.

This section will outline two sample queries on the dialog annotation and then discuss their implementation in the MATE and AG query languages.

Example Queries

Two sample queries have been chosen to illustrate the kind of question that might be put to an annotated dialog and also to highlight weaknesses in the two query languages under review. The queries are:

Q1. Find pronouns that are separated from their antecedent by more than 3 dialog turns.

Q2. Find dialog turns with an flf label 'info-request' and list the phrase accent/boundary tone sequences of the associated intonationally defined utterances.

The MATE Query Language

Queries in the MATE query language (Q4M) (McKelvie et al., 2000) are made up of an initial set of definitions which establish variables quantified over certain labels, for example ($u = utterance) establishes a variable $u which stands for any utterance in the corpus. This is followed by a series of conditions on the tokens and their attributes. Attributes are referenced using a dot notation, for example the part of speech (pos) attribute of a word would be referred to as $w.pos. Q4M contains a wide range of comparison operators which can be used to compare both attribute values and the temporal and structural relations between tokens.

The first query can be expressed as in figure 2, left. Here we have assumed that the relationship between the pronoun and its antecedent can be verified by comparing attribute values; however, there may be a more direct test in the language although this is not apparent from the available documentation. The constraint on the relative position of the two utterances (more than three turns apart) is realised by comparing their id attributes. This special attribute is maintained by MATE for just this purpose.

This query gives exactly the results required. The pronouns matching the defined constraints will be reported in the query results. Q4M returns results in a form which allows the original tokens to be identified in the corpus; one result of this is that further queries are possible on the result of one query. We could, for example, find the pitch accents associated with these pronouns via an additional query.

The second query might be expressed as in figure 2, right. Here the four variables range over different kinds of token and the additional conditions constrain the word to be part of both the utterance and the intonational phrase. The final condition constrains the tone to be temporally included within the

intonational phrase. The result of this query then is a collection of bindings for these variables which make these conditions true.

Although this query seems to express our intended question, the result is not what is required since each result set identifies only one word and tone rather than the set of words and tones associated with the intonational phrase. Q4M provides no way to apply a condition to a set of tokens, such as the set of words or tones included in the intonational phrase.

The AG Query Language

In a recent paper, Bird and others (Bird et al., 2000) propose a query language for the annotation graph data model based on describing paths through the acyclic annotation graph.

Queries in the AG query language are made up of a set of path expressions which describe paths through the annotation graph. For example in the query term X.[:utt, flf:  assert].Y, X and Y stand for nodes in the annotation graph and the terms in brackets specify the type (:utt) and attribute values associated with the arc between these nodes.

```
select token(P) where
 X.[:utt].V.[:utt]*.W.[:utt].Y
 W.*[:word, label: P,
            pos: pronoun,
            antecedent: A]*.Y
 X.*[:word, referents: A]*.V
```

```
select
  F(X1).[:inton id: F(I1), label: L1].F(X2),
  F(V).[:tone id: F(I2), label: L2].F(W)
where
  X.[:utt, flf: info-request].Y
  X.[:word]*.Y
  X.[:inton]*.X1.[:inton, id: I1, label: L1].X2.[:inton]*.Y
  V.[:tone, id: I2, label: L2, contained(I1)].W
```

Figure 3: The example queries expressed in the proposed AG query language.

The first query can be expressed as in figure 3. This query can be paraphrased as: there is a sequence of at least three utts between X and Y such that the first ends at V and the last begins at W; there is a pronoun between W and Y with antecedent A; there is a word between X and V which has an attribute referent with the value A.

Note that in order to express the condition on the distance between utterances, we needed an explicit pattern with that many utterances. While this is manageable for a sequence of three it would become cumbersome for larger sequences. The addition of a quantified repetition (kleene-*) operator, for example X.[:utt]*{3,}.Y, would address this problem.

The result of this query is a set of bindings for the variable P being the labels on the pronouns matching the query. Other information (such as a unique identifier) could be returned by specifying additional variables in the select part of the query.

The second query is not straightforward in this query language since it too lacks a method for applying a relational condition to all elements in a repeated sequence. A method for answering this query has been suggested (Bird, private communication) which involves a query language extension for constructing new annotations from query results. The example shown in Figure 3 constructs a new annotation structure containing intonational and tone segments matching the stated criteria. The function F() is a skolem function and ensures that identifiers are not duplicated in the new annotation structure. Each instance that matches the query produces a new intonational and tone token; however, since the duplicate intonational tokens are made identical via the skolem function, the result will be a set of intonational tokens dominating sets of tone tokens.

The addition of this facility to the annotation graph query language makes possible a powerful new mode of operation: building new annotations from existing ones. This kind of operation is available in the Emu system by writing Tcl scripts but is generally used only by programmers because it requires knowing the Tcl language. The complexity of the AG query language is such that complex queries may be too difficult for end users to compose.

DISCUSSION

The creation of an example dialog annotation using DAMSL and ToBI has highlighted a number of issues for annotation system design, in particular in the design of a query language for this domain.

The annotation structure used on the dialog is largely a good fit to the three data models under review with the exception of the need for multiple attribute values on speaker turns. This problem would be relatively easy to overcome in any of the models but requires support at a low level to allow full flexibility in querying multiple valued attributes.

The query languages for the MATE and Annotation Graph (AG) systems were both able to express our first test query although the AG query language makes the expression of this query somewhat longwinded as it lacks a quantified repetition operator.

The second query required applying a condition over all of the members of a sequence of tokens; neither query language provides a construct to express this condition. The AG query language allows the construction of a new annotations based on query results. While this feature allows much more complex queries to be answered, it requires a level of understanding of annotation system internals which may not be available to Linguists and other corpus users. If this kind of query operation is to be used by this group, work must be done on simplifying the user experience either with a simpler query syntax or via a graphical user interface.

In general the MATE query language is much easier to read and understand than the AG query language since it consists of simple pairs of terms expressing conditions on a set of tokens defined in the preamble. It is likely that a more readable surface form might be developed for the AG language since, at present, its development has concentrated on it's formal properties rather than usability. The AG query language also needs to simplify the expression of hierarchical paths through the annotation; the current proposal only provides a mechanism for expressing conditions on sequential paths; hierarchical queries must be expressed as patterns matching the implicit hierarchical structure of the annotation graph. Many queries are naturally expressed as a hierarchical condition and so for usability, any query language should make these conditions explicit.

ACKNOWLEDGEMENTS

References

Allen, J. and Core, M. (1997). Damsl: Dialog act markup in several layers. available at: `http://www.cs.rochester.edu/research/trains/annotation/RevisedManual/Re%visedManual.html`.

Bird, S., Buneman, P., and Tan, W. C. (2000). Towards a Query Language for Annotation Graphs. In Proceedings of LREC 2000, Athens, Greece.

Bird, S. and Liberman, M. (2000). A Formal Framework for Linguistics Annotation. Speech Communication. To appear in a Special Edition on Annotation Tools and Systems.

Cassidy, S. and Bird, S. (2000). Querying Databases of Annotated Speech. In Orlowska, M. E., editor, Proceedings of the 11th Australasian Database Conference, volume 22 of Australian Computer Science Communications, pages 12–20, Canberra, Australia. IEEE Computer Society.

Cassidy, S. and Harrington, J. (1996). EMU: an Enhanced Hierarchical Speech Data Management System. In Proceedings of the 6th International Conference on Speech Science and Technology, pages 361–366, Adelaide.

Cassidy, S. and Harrington, J. (2000). Multi-level annotation in the emu speech database management system. Speech Communication. To appear in a Special Edition on Annotation Tools and Systems.

McKelvie, D., Isard, A., Mengel, A., Grosse, M., and Klien, M. (2000). The MATE Workbench - an annotation tool for XML coded speech corpora. Speech Communication. To appear in a Special Edition on Annotation Tools and Systems.