Annotation Scheme for Authored Dialogues
Version 1.0

Svetlana Stoyanchev and Paul Piwek
Centre for Research in Computing, The Open University, Milton Keynes, UK
{s.stoyanchev, p.piwek} @open.ac.uk

Contents

Chapter 1

Introduction

The annotation we describe here has been developed in the context of the CODA project[1]. The goal of this project is to develop tools and technologies for automatically generating dialogue from monologue. The idea is to create a corpus of aligned monologue and dialogue and automatically derive rules from this corpus for converting monologue into dialogue. Thus a first step towards the overall goal is to create a parallel corpus of monologues and dialogues that express the same information. To create the corpus, we start from authored (i.e., fictive) dialogues and want to:

1. segment the dialogue into dialogue acts (questions, answers, explanations),
2. label segments with one or more dialogue act types, and
3. align the dialogue with a monologue that expresses the same information. To do this we group the segments from the dialogue, such as Question/Answer pair or an explanation and convert it into a monologue.

Much of the previous research on dialogue modelling focused on spoken task-oriented dialogue. In this work we look at a different kind of dialogues – authored dialogue in fiction and philosophy, such as Mark Twain's "What is Man" [9]. We are interested in capturing how authored dialogues present information, make arguments, and which dialogue acts are used by characters in such dialogues.

The form and purpose of authored dialogues differs from naturally-occurring spoken dialogues. For example, if an authored dialogue contains a clarification, its purpose usually is to clarify information for the reader (rather than the interlocutors). Also, according to our preliminary corpus analysis, most of the clarifications

[1] See [13] and computing.open.ac.uk/coda.

in authored dialogues are on the level of consideration (level 4 of Clark's theory [5]).

We develop a new annotation scheme for authored dialogues. In our annotation scheme we are interested to capture how authors present arguments to the readers through the interaction between the dialogue characters. In the following sections, we review previously developed dialogue annotation schemes and describe our new scheme. This scheme is designed with the purpose of annotating and translating authored dialogues.

Chapter 2

Related work

Researchers have proposed several theories for analysing dialogue. Searle [12] proposed a taxonomy of Illocutionary Acts which classifies an utterance as either assertive, directive, commissive, expressive, or declarative. More recently, Carletta et al. [3] proposed a scheme that captures deep dialogue structure. Dialogues are analysed on three levels: 1) dialogue moves, 2) dialogue games, and 3) dialogue transactions. The scheme has been used to annotate the Maptask [1] corpus of human-human task-oriented communication.

Bunt [2] proposed Dynamic Interpretation Theory (DIT). The main goal of DIT is to provide a computational model of communicative agents. This scheme emphasises that the agents in dialogue aim at the achievement of a communicative goal and each utterance is a contribution towards this goal. The scheme describes dialogue control management including social obligations management, interaction management, and feedback, as well as task-oriented communicative functions.

Core and Allen [6] designed a generic scheme for dialogue analysis (DAMSL). The scheme allows assignment of multiple dialogue acts to a single utterance. The authors point out that an utterance in a dialogue can play multiple roles and have hidden meanings: "an utterance may simultaneously accept information and acknowledge that the information was understood as well as answer a question". Task-oriented human-human (Trains corpus) as well as non-task-oriented human-human dialogues (Switchboard corpus) were annotated with a variation of the DAMSL scheme.

We find a lot of similarities between the previously proposed schemes. For example, the dialogue move level annotations of Carletta et al.'s scheme closely correspond to the Forward and Backward communicative functions of DAMSL scheme and task-oriented communicative functions of DIT. Table 2 shows the correspondence between Carletta's, DAMSL, and our tag set.

DAMSL	Carletta	CODA
forward; statement; assert	init; statement (explain)	Init-Explain
forward; statement; reassert		
forward; other-statement;		
forward; directive; info-requets	init; question; Wh	Init-Factoid-InfoReq, Init-Complex-InfoReq
	init; question; YN	Init-YNQ-InfoReq
forward; directive; action-directive	init; command; instruct	Init-Instruct
forward; commit speaker future action		
forward; performative		
backward; agreement; accept		Resp-Agree
backward; agreement; accept-part		Resp-Agree
backward; agreement; accept-maybe		Resp-Agree
backward; agreement; reject		Resp-Contradict
backward; agreement; reject-part		resp-contradict
backward; agreement; hold		
backward; understanding; signal-nonunderstanding	resp-communicate-ack	Init-Clarify-Request
backward; understanding; signal understanding; acknowledge		Resp-Acknowledge
backward; understanding; signal understanding; repeat-rephrase		
backward; understanding; signal understanding; completion		
backward; understanding; correct misspeaking		
backward; answer	resp-info-yes resp-info-no resp-info-wh resp-info-clarify	Resp-Answer-Yes Resp-Answer-No Resp-Factoid Resp-Explain
backward; information-relation		

Table 2.1: Comparison of Dialogue Act tags from DAMSL, Carletta et al., and CODA annotation schemes. DAMSL and Carletta et al. allow multilevel tagging. ';' separates tags on different levels.

Most of the previous work focused on task-oriented dialogues. The previously proposed dialogue schemes were developed in the context of projects aiming at conversational agents for human-computer interaction or interpretation of human-human communication. In our work, we deal with fictive expository dialogues whose purpose is to present an idea/information to the reader. While previous schemes modelled communication between two agents, we are modelling the flow of information between two agents and its effect on the reader. In particular, we are interested in determining how dialogue acts map to rhetorical structures in monologue.

Expository dialogues are written by a single author. Since they are meant to resemble real dialogues, we borrow most of the dialogue act tags from DAMSL and Carletta et al. adapting them to our purpose. Even though the characters in an authored dialogue do not have a real human-human dialogue, we do occasionally find clarification dialogues in our corpus. These are typically included by the authors to emphasise a particular point (see [10]). We do not include DAMSL's communication level tags in our schema because these clarifications are infrequent in our corpus. We allow multiple level of annotations, as proposed by DAMSL to a limited degree. We find that in our corpus, if an utterance carries multiple functions, it can generally be split into segments that each carry a different function. For example, *yes, the mind ...* can be split into a positive answer *yes* and an explanation *the mind ...* which is an explanation dialogue act. For the cases when an utterance is both a response and an initiation, we allow multiple tags: the annotator can assign both a primary and (if required) a secondary tag to an utterance.

Chapter 3

Annotation Guide

3.1 Definitions (alphabetically ordered)

- **Decorative** information is included by the dialogue author to liven up the dialogue and create natural transitions between segments (see definition below) of key information.
- **Informational Unit (IU)** A set of segments which together translate to a single *snippet*. An IU should satisfy the following constraints:
  - An IU should not contain a proper subset of segments which could also translate to a snippet. In other words, IUs should be as small as possible.
  - Every segment in a dialogue should belong to exactly one IU. Because each IU corresponds with exactly one snippet, every segment in a dialogue maps to exactly one snippet.
  - If two segments cannot be translated into a monologue snippet independently of each other, they should belong to the same IU.

  Examples of IUs are:
  - An init dialogue act followed directly be a response dialogue act.
  - A single dialogue act, typically a key *init-explain* or *resp-explain* act.

  The following shows a schematic example of a dialogue that has been partitioned into two IUs. The two question-answer pairs each constitute an IU.

1. A: [Question (init)
2. B: Answer (response)]$_1$
3. [Question (init)
4. A: Answer (response)]$_2$

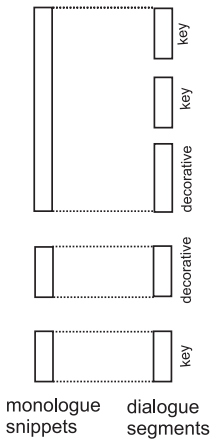

Figure 3.1: Schematic example of aligned monologue snippets and dialogue segments. Note that the first snippet aligns with three segments, which together constitute an Informational Unit (IU)

- **Key** information in an expository dialogue contributes to the main information/ideas that the author is trying to get across; it is directly relevant to the main purpose of the dialogue. It should be present in the corresponding monologue. Generally, key information contains concept words.

- **Monologue Sentence** Sentences in monologue should be marked by punctuation. Sentence boundaries do not need to correspond to snippet boundaries.

- **Segment**: substring of a turn that is either labelled key or decorative information. A segment of key information typically consist of a dialogue act (or two dialogue acts in case the segment expresses at the same time a forward and a backward looking dialogue act). Each substring of a dialogue should be part of exactly one segment. If two consecutive segments of the same speaker have the same dialogue act and map to the same monologue snippet, they should be merged.

- **Snippet**: a monologue translation of a set of segments which together form an *informational unit*. Snippet boundaries should coincide with elementary

discourse unit boundaries (EDUs) of the rhetorical structure of the text – each snippet boundary should be a EDU boundary, *but* not each EDU boundary has to be a snippet boundary.

- **Turn** : everything dialogue participant *A* says before dialogue participant *B* takes over.

3.2 Using the CODA Annotation Tool

3.2.1 Creating New Annotations

1. Open a dialogue file: Dialogue — open
2. Tag several turns
   (a) Click on a cell in Tagged Turn column.
   (b) Highlight a segment (part of a Turn).
   (c) Select a Type of the segment(Refer to Section 3.3).
   (d) Select a Dialogue Act of the segment (Refer to Section 3.4).
   (e) Click *add* to add a segment to the segments that have been identified and tagged so far.
   (f) Repeat above steps until the turn has been fully segmented and click *save* to save the annotation of this turn.
3. Write a monologue snippet:
   (a) Identify a set of segments that together form an Information Unit (IU).
   (b) Enter the ids of these segments into the "Mapping index" text box (e.g., "1 4"; ids are separated by a space).
   (c) Click *Insert_segments*: the mapped segments are copied into the snippet editing window.
   (d) Edit snippet's text in the snippet editing window to create a fluent monologue text.
   (e) Click Add to insert a monologue snippet.
4. Save dialogue annotations: File | save dialogue
5. Save monologue annotations: File | save monologue

3.2.2 Editing Annotations

Changing dialogue act

To change a dialogue act of a segment or to add a secondary dialogue act tag use one of the methods:

1. Edit the XML file directly in a text editor.
2. Open the dialogue in the D2MTool. Click on the segment that needs to get a tag change. Change the tags, click add/change button. The change should appear in the segment display window. Click save.

Adding a segment

To add a segment, the D2MTool should be used (to preserve the mapping of monologue snippets to the dialogue segments):

1. Open the dialogue in the D2MTool,
2. Open corresponding monologue in the D2MTool,
3. Click on the turn that needs further segmentation
4. Remove or add segments
5. Click save to save the turn segmentation. The new segment is not mapped to any of the dialogue snippets. All the previous mappings are preserved (segments following the new segment get their ids incremented)
6. To map the new segment to a new monologue, select a monologue snippet before the new snippet, follow the steps described above to add a new snippet. Alternatively, edit XML in a text editor.

Modifying monologue

Edit XML directly or use D2M Tool: select the monologue snippet, click on view/edit, edit the text, click save.

3.3 Key and Decorative

In a dialogue, the interlocutors exchange information with each other. Information that is directly relevant to the main purpose of the dialogue is classified as *key information*. For instance, in a dialogue which consists of a discussion about some

topic (say, whether holes exist as material objects), information that presents either side of the argument is key information. Only key information is translated into monologue.

Decorative segments perform what Bunt calls dialogue control functions. This includes social talk (phatic communication) and meta-communication. Decorative segments are not translated into monologue.

Most utterances in an authored dialogue present *key information*. They are about the topic of the dialogue and their meaning needs to be preserved in the monologue. They can be copied verbatim to the monologue or paraphrased. There are, however, also utterances which do not contain key information; in terms of Dynamic Interpretation Theory (DIT) [2] they express 'dialogue control functions'. From the point of view of an author of fictional dialogue they are often used to create a certain effect in the audience: e.g., the creation of a mood, attracting attention of the reader, or embellishing a dialogue. Such utterance express *decorative information*. Examples of decorative utterances are utterances for managing turn taking such as *Wait!* or *Just a moment*. Decorative segments also include exchanges which concern the *social context* of the dialogue. For example, decorative social dialogue may be found at the beginning or end of a fictional dialogue where characters establish acquaintance or say farewell. Decorative utterances can also occur in the body of a dialogue (as opposed to the beginning and end).

To test if a segment is key or decorative, answer the question: Can the information of the segment be omitted, without affecting the interpretation of the key information? If yes, the segment is decorative,otherwise it is key.

A decorative segment is underlined in the following example:

A: *For instance ?*
B: *Well, then, for instance.*
B: *Take the case in the book here ...*

Another example of a decorative statement is when one speaker praises the other:

Good job; Well done, etc.

3.3.1 Primary and Secondary Tags

In the DAMSL annotation scheme [6] the authors note that that an utterance in a dialogue may carry multiple communicative functions. An utterance can be an answer to a question, and at the same time it may be initiating a new idea. The authors differentiate between possibly co-existent *Forward* and *Backward* communicative

functions of an utterance. Our annotation scheme allows multiple dialogue acts, similarly to DAMSL. However, to simplify the annotator's job we limit the allowed tag values.

The annotators of authored dialogues are required to annotate primary function and have an option of annotating secondary function. The values available for the secondary function of a segment depend on the value chosen for the primary function of this segment. For example, if a primary function of a segment is *Init-Explain*, the only available secondary functions are *Response-Agree* and *Response-Contradict*. Alternatively, we could have required annotators to choose one value from *Forward* and/or *Backward* function which would yield the same end-result. Our choice was motivated by the goal of reducing the mental strain on annotators.

3.4 Dialogue Act Annotation Procedure

Segment an utterance such that each *key* segment has a single primary dialogue act. Then assign primary and secondary (when applicable) moves to the segment. If two consecutive segments in a single turn have the same primary and secondary move, merge them.

1. If the primary purpose of a segment is to initiate an Informational Unit (IU)[1], assign the most appropriate *init* primary tag. Usually but not necessarily:
   - An answer/response to the segment follows in the next turn.
   - This is the last segment of a turn.
2. If the primary purpose of a segment is to respond to a previous utterance, assign the most appropriate *response* primary tag. Usually but not necessarily:
   - The next speaker's utterance is not a response to the current utterance.
3. If the primary tag of the segment is *Init-Explain, Init-YNQ, Init-ComplexQ*, and this segment also serves as a response (agreement or contradiction) to the previous utterance, assign the secondary tag *Resp-Agree* or *Resp-Contradict*.
4. If the primary tag assigned is *Resp-Agree, Resp-Contradict, Resp-Explain*, and if the utterance is followed by a response or an answer, assign a secondary tag of *Init-Explain*.

[1] For our purposes, we define an IU generally is a init-response adjacency pair.

3.5 Choosing a Dialogue Act

Figures 3.2 and 3.3 illustrate the taxonomy for Initiating and Responsive dialogue acts.

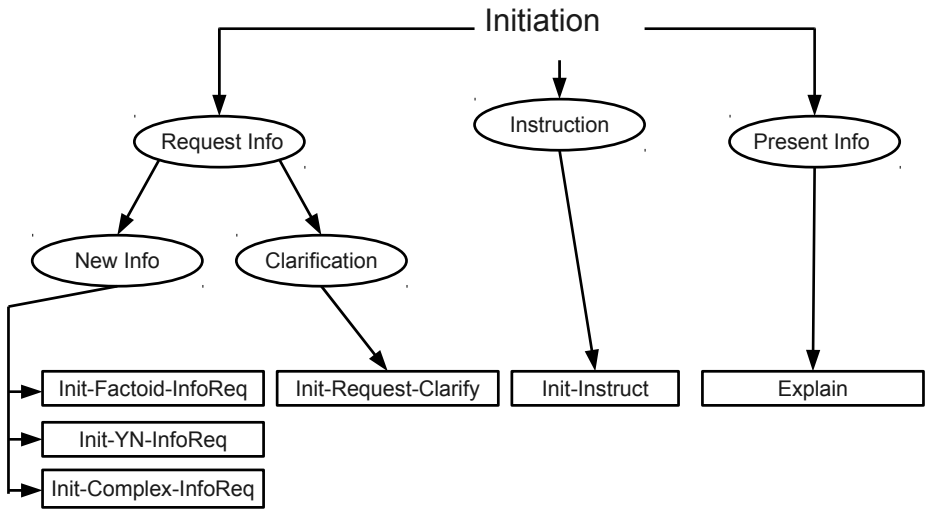

Figure 3.2: Taxonomy of initiating dialogue acts.

3.5.1 Initiating Dialogue Acts

Initiating dialogue acts correspond to DAMSL's *forward* looking communicative function and to *Initiation* moves of Carletta et al.

Init-Explain

Init-Explain dialogue act states key information that also should be presented in the corresponding monologue. The act may describe an opinion, for example:

The mind is independent of the man.

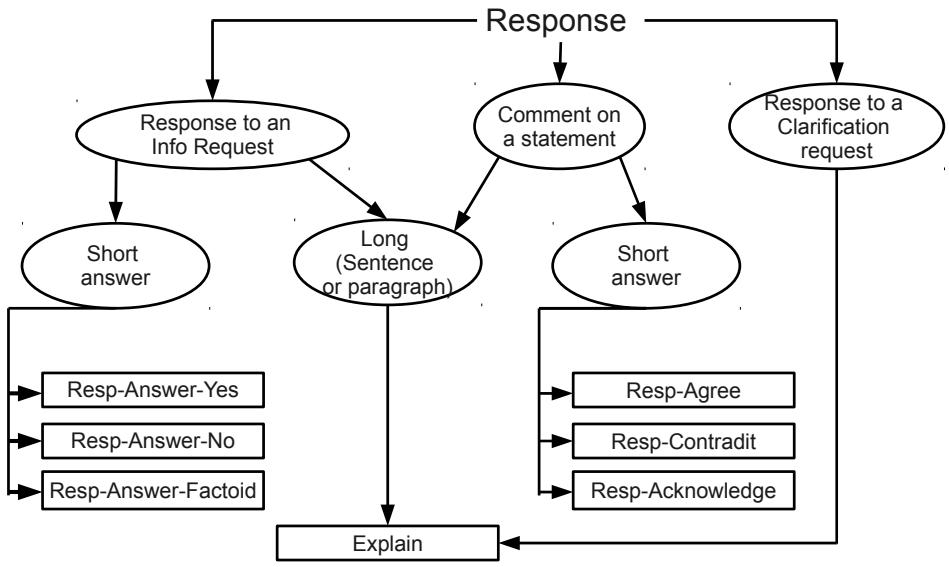

Figure 3.3: Taxonomy of responding dialogue acts.

It may also be an example given by one of the characters, such as:

For example, the algorithm may be written in a high level language that allows, say, multiplying integer matrices in one step.

Init-Explain can be a long on-going explanation:

The others offer your a hundred bribes to be good, thus conceding that the Master inside of you must be conciliated and contented first, and that you will do nothing at FIRST HAND but for his sake; then they turn square around and require you to do good for OTHER'S sake CHIEFLY; and to do your duty for duty's SAKE, chiefly; ...

Assign this act if all of the following are true:

- This segment is a statement, a set of statements (it may contain rhetorical questions as well).
- The primary purpose of the segment is to initiate a new interaction.
- The segment contains key information: if this segment is not present in the monologue, some important information is lost.

In most cases, *Init-Explain* is followed by a response (agreement, contradiction, clarification), however it is possible that an initiating segment is not addressed or acknowledged right away (or at all).

Possible secondary dialogue acts are *Resp-Agree* and *Resp-Contradict*. Below, the underlined segment gets a primary tag of *Init-Explain* because it starts a new idea. It gets a secondary tag of *Response-Agree* because it supports the yes answer to the previous question, so it also serves a *backward looking* function:

A: *And that his mind works automatically and is*
independent of his control – carries on thought on its own hook?
B: *Yes* . It is diligently at work , unceasingly
at work, during every waking moment. If it needed the man's help
it would wait for him to give it work when he wakes in the morning.
A: *Maybe it does.*

Init-Factoid-InfoReq

Assign this tags if all of the following are true:

- The segment is a question or an imperative such as "Tell me ..." that requests information (who, when, where, what, how many, etc.)
- It can not be answered with yes/no
- The most natural answer to the question is a simple factoid answer (and not elaboration or explanation), such as a name or a number. Although sometimes the character may elaborate on factoid questions, hence do not use the actual answer to judge.
- The segment is not why or how question.
- If it is a *what* question, it does not ask about the complex issues, such as cause, effect, etc.

A possible secondary dialogue act is *Init-Request-Clarify*. Assign this secondary dialogue act when the factoid question is asked with an intention of clarifying previous utterance.

Init-Complex-InfoReq

Assign this tag when all of the following are true

- The segment is a question.
- The segment is not a factoid question.
- The segment is not a Yes/No question.
- The segment is not a rhetorical question. [2]

Init-YN-InfoReq

Assign this tag when all of the following are true

- The segment is a question
- The segment can be answered with yes or no.

Possible secondary dialogue acts are *Response-Agree*, *Response-Contradict*.

[2] Rhetorical questions are annotated with *explain*

Init-Clarify-Request

Clarification request always playes dual role of an information request as well as a response to previous utterance.

A speaker asks the other speaker to clarify, for example:

How do you mean?

Explain

Assign this tag when all of the following are true

- A segment is either a statement or a question that illicits clarification or additional information.
- If this segment is not present in the monologue, no important information is lost. [3]

Possible secondary dialogue act is *Init-FactoidQ* or *Init-YNQ*.

3.5.2 Responsive Dialogue Acts

Responsive dialogue acts correspond to DAMSL's *backward* looking communicative function and to *Response* moves of Carletta et al.

Response-Answer-Yes

A positive answer to a yes/no question:

Yes!

Oh yes, there is.

it is true.

Perhaps so.

Assign this statement when:

- Previous dialogue act is *Init-YNQ*

[3] If a segment asks for a clarification and carries information, it should be split into two segments.

- The segment can be replaced by 'Yes' without the loss of information

This act is likely to be followed by *Explain* dialogue act:

A: *Oh yes, there is. It is candidly stated, this time. That has not been done before.*
B: *Well, yes, that is a difference, it is true.*

Possible secondary dialogue acts are *Init-Explain* or *Response-Explain*. Consider an example

A: *Is it raining?*
B: *Cats and dogs*

B's response is a positive answer to a yes/no question as well as an explanation. It should be marked as *Resp-Answer-Yes* as a primary tag and *Resp-Explain* as a secondary tag.

Response-Answer-No

A negative answer to a yes/no question. Assign this tag when:

- Previous dialogue act is *Init-YNQ*
- The segment can be replaced with "no" without loss of the information

Possible secondary dialogue acts are *Init-Explain* or *Response-Explain*. Consider an example

A: *Is it raining?*
B: *It is snowing*

B's response is a negative answer to a yes/no question as well as an explanation. It should be marked as *Resp-Answer-No* as a primary tag and *Resp-Explain* as a secondary tag.

Response-Agree

A statement that shows an agreement with the previous statement, for example

You are right. It is true.

Assign this tag when:

- Previous dialogue act is *Init-Explain*
- The segment indicates agreement or support to the previous utterance

There is an overlap with *Response-Yes* dialogue act. The main difference is that *Response-Yes* is preceded by a yes/no question, while *Resp-Agree* is preceded by a statement.

A possible secondary tag is *Init-Explain*. In the following example, B both agrees with A and initiates an explanation (although it is questionable which tag should be primary and whicha secondary):

A: The air is warm
B: It is warm and humid.

Response-Contradict

A statement that shows a disagreement or contradiction with the previous statement, for example

I don't know where you got it from
This can not be true

Assign this tag when:

- Previous dialogue act is *Init-Explain*
- The segment indicates disagreement or contradiction with previous utterance

Possible secondary tag is *Init-Explain*.

Response-Acknowledge-Neutral

Example:
Maybe it does.

Assign this tag when:

- Previous dialogue act is *Init-Explain*
- The segment indicates acknowledgement of the previous statement without clear agreement or contradiction

Possible secondary tag is *Init-Explain*.

Response-Factoid

Assign this tag when:

- Previous dialogue act is *Init-FactoidQ*
- The segment is a simple factoid answer to the question.
- The segment contains key information

Response-Explain

An extended answer to a question, response to a command, elaboration on an agreement or disagreement.

Perhaps so. The same advantage he might get out of thinking himself a duke, and living a duke's life and parading in ducal fuss and feathers, when he wasn't a duke at all, and could find it out if he would only examine the herald's records.

Assign this tag when the segment contains key information and one of the following applies:

- The segment is an answer to the question in preveous turn
- The segment is a comment to the statement in the previous turn
- Previous segment in the same utterance is marked as *Resp-Yes, Resp-No, Resp-Factoid* and previouse utterance contained a queston.

Possible secondary tag is *Init-Explain*.

NOTE: It is common for an *Explain* move to be both Init and Response.

3.5.3 Transition Tags

Transition tags are the segments that link together the presentation of an idea into a coherent dialogue. Transitions are by definition decorative. They are part of dialogue but not the equivalent monologue. If a transition segment is removed, no information is lost. So, transitions are special types of decorative segments. We identify two types: Pause and Praise.

Trans-Pause

Segments that indicate a pause

At any rate; Anyway; Well; So; etc.

We are interested in capturing these to be able to generate the dialogue from a monologue.

Other

If none of the tags applies, a segment is annotated as *other*.

3.6 Example

Consider an example:

A: *At any rate , he can make it stick to a subject if he wants to*
B: *Not if it finds another that suits it better. As a rule it will listen to neither a dull speaker nor a bright one. It refuses all persuasion.*

Segmentation and annotations of the above utterances:

1. At any rate **decorative**
2. he can make it stick to a subject if he wants to **key, Init-Explain**
3. Not if it find another that suits it better **key, Resp-Contradict**
4. As a rule it will listen to neither a dull speaker nor a bright one. It refuses all persuasion. **key, init-explain**

The first segment is not converted into a monologue, it is a decorative statement used by the speaker A in order to emphasise his level of commitment, bring attention to the utterance. Segments 2, 3, and 4 are *key* as they carry information that will be present in the corresponding monologue.

Monologue sentence paraphrase:

> *He can make it stick to a subject, but not if it finds another that suits it better.*

Monologue snippets and their mapping:

1. he can make it stick to a subject **segment 1, 2**

2. but not if it finds another that suits it better **segment 3**

3. As a rule it will listen to neither a dull speaker nor a bright one . It refuses all persuasion **segment 4**.

Chapter 4

Discourse Annotation

4.1 Existing Annotated Corpora

Most large scale discourse annotation efforts have focussed on news articles, typically of the sort found in the Wall Street Journal (Carlson et al. [4], Wolf and Gibson [14], Penn Discourse Treebank [11]). Whereas most of this work has assumed that the structure of annotations is tree-shaped, Wolf and Gibson have departed from this assumption, allowing crossing dependencies. The number of relations that is used varies widely: e.g., 78 were used by Carlson et al. (grouped into 16 more general types of relation), three levels were used for the PDTB (4 relation at the most general level, 16 at the level below that, and 29 on the most specific level), and the HILDA discourse parser [7] uses 18 relations (16 from Marcu's general relations + Same-Unit and Textual-Organisation). Note that the work on the PDTB differs from the other efforts by focussing on disambiguating/interpreting the meaning of connectives, rather than uncovering (possibly implicit/inferred) relations in a text.

4.2 Annotation Procedure

In this section we describe the process for annotating Rhetorical structure on the manually written monologues. For the M2D translation project we perform *partial annotations*. For the purpose of the M2D translation some of the relations inside the monologue are irrelevant: relations inside the snippets that are mapped 1-to-1 to a dialogue segment and copied verbatim will not be present in the M2D rules. We call these snippets multiple discourse units (MDU) and do not split them further. Identifying MDUs allows us to save time on annotation while annotating all discourse relations that contribute to the M2D transformation rules. The relations

Marcu general	Marcu specific	Penn Treebank general	Penn more general	Penn specific
Attribution	attribution, attribution-negative			
Background	background, circumstance			
Cause	cause, result, consequence	cause	reason, result	contingency
Comparison	comparison, preference, analogy, proportion			
Condition	condition, hypothetical, contingency, otherwise	Condition	hypothetical, general, unreal/factual present/past	Contingency
Contrast	contrast, concession, antithesis	Contrast	juxtaposition, opposition	comparison
Elaboration	elaboration-additional, elaboration-general-specific, elaboration-part-whole, elaboration-process-step, elaboration-object-attribute, elaboration-set-member, example, definition			
Enablement	purpose, enablement			
Evaluation	evaluation, interpretation, conclusion, comment			
Explanation	evidence, explanation-argumentative, reason			
Joint	list, disjunction			
Manner-Means	manner, means			
Summary	summary, restatement			
Temporal		Synchronous Asynchronous	precedence, succession	temporal temporal
Topic-Change	topic-shift, topic-drift			
Topic-comment	problem-solution, question-answer, statement-response, topic-comment, comment-topic, rhetorical-question			
Same-Unit				
Textual-Org				

Table 4.1: General and specific relations defined by Marcu and corresponding Penn Treebank relations.

inside an MDU are internal segment relations and will not be part of translation.

4.2.1 Segmentation

- Automatically segment monologue into EDUs using monologue snippets from the annotation.
- For each automatically identified EDU, if it is not an MDU, segment it further: create an EDU boundary where Marcu's segmentation rule applies.

For example, the following four segments of a dialogue

1. A: key:Init-YN-InfoReq:none: Did the man possess it who gave the old woman his last shilling and trudged home in the storm?
2. B: key:Resp-Explain:none: He had the choice between succouring the old woman and leaving her to suffer.
3. key:Init-YN-InfoReq:none: Isn't it so?
4. A: key:Resp-Answer-Yes:none: Yes,

are translated into a single monologue snippet:

> *The man who gave the old woman his last shilling and trudged home in the storm had the choice between succouring the old woman and leaving her to suffer.*

As the monologue snippet is not an MDU, during manual annotation it is split as indicated by the "|":

The man | who gave the old woman his last shilling and trudged home in the storm | had the choice between succouring the old woman and leaving her to suffer.

The goal of manual segmentation is to 1) add segment boundaries inside snippets in order to define relations between them and 2) fix the text of the monologue snippet 3) fix the existing boundaries of the snippets.

To add a new boundary, simply add it.

RST annotation is an opportunity to improve manual translation of dialogue to monologue by finding mistakes, inconsistencies, and style of the monologue. If you see a mistake or something that may be expressed in a better way, fix it here.

Fixing boundaries is necessary in rare cases when the snippet boundary mistakenly crosses an EDU boundary remove the boundary and if necessary add it in a correct location.[1] In the following example, the snippet boundary crosses an EDU:

[1] This hardly ever happens and arises from incorrect segmentation during dialogue-to-monologue translation, as the annotators were instructed to create snippets matching EDU boundaries.

I would not use those words – Free Will – but others: | Free Choice
To fix the segmentation, remove the boundary before *Free Choice* and add a boundary before *but* for a contradiction relation between the two segments:
I would not use those wosrds – Free Will –| but others : Free Choice

4.2.2 RST Annotation

In our annotation scheme we were guided by the annotation scheme defined by Carlson et al. [4]. These relations and their groupings defined by the authors are summarized in Table 4.1. We use the grouped list as the basis for the annotation scheme. We choose to expand the most frequent relations in our data (Evaluation, Elaboration, Explanation, and Topic-comment) into their sub-relations. For these relations the annotator has an option of selecting a fine-grained tag or high level tag. This decision was inspired by the method of Penn Treebank annotations where annotators chose granularity of their labels. An additional advantage to using fine-grained tags for the above relations is that it will force annotators to think more thoroughly and disambiguate between explanation, evaluation, and elaboration, the three most common and most confusable relations.[2] We did not expand the rest of the relation tags because 1) they are not as frequent 2) the semantics of the sub-tags is closer to each other. Some of the tags were merged or renamed as indicated in the footnotes. The new names better represent the meaning of the tag. The following tags are used in the discourse annotation of CODA corpus:

1. Explanation (S explains N)
    - Evidence (Evidence is satellite)[3]
    - Reason (Reason is satellite)
2. Enablement (S enables N)
3. Cause (S causes N)[4]
4. Evaluation (S evaluates N)
    - Subjective[5]
    - Inferred[6]

[2] According to our initial observation
[3] Merged Carlson et al. **??** Evidence and Explanation-Argumentative
[4] This order was changed form Marcu's definition for consistency and ease of annotation purposes
[5] Corresponds to Marcu's Assessment and Interpretation
[6] Corresponds to Marcu's Conclusion

- Comment (Comment is satellite)

5. Attribution (S is the person, organization, etc. to whom N is attributed)
6. Condition-Hypothetical (S is condition for N)
7. ContrastMono (for concession and antithesis), or Contrast for multinuclear contrast relation
8. Comparison
9. Summary (S summarizes N)
10. Manner-means (S is meansor manner of achieving N)
11. Topic-Comment
    - problem-solution (Solution is satellite)
    - Statement-response (Response is satellite)
    - Question-Answer (Answer is satellite)
    - Rhetorical Question (Question is satellite)
12. Background (Background info is satellite)
13. Temporal (when S, N)
14. Elaboration
    - Additional (Additional information is satellite)
    - General-Specific (specific information is satellite)
    - Example (Example is satellite)
    - Object-attribute (Attribute is satellite)
    - Definition (Attribute is satellite)
15. Same-unit (multinuclear)

Marcu et al. [8] notes that the "In some cases, more than one relation may hold between two textual segments. For example, a causal and a temporal relation may hold between two segments simultaneously". The authors determine a protocol that defines the order of applying the relations according to their specificity. More specific relations are applied first. Relations in the list above are ordered from more to less specific and should be applied in this order.

4.3 Scoping

To determine scope of rule application, use deletion rule [8] (page 34).

4.4 Modifications to the Annotation Guidelines

This section defines CODA modifications to RST tool annotation procedure and Marcu's annotation guidelines.

4.4.1 Annotation Depth

We define *relation depth* as the distance from the node to its most distant child When capturing D2M rules we are interested in determining relations between spans of text. We find that our inter-annotator agreement on RST annotations is higher for more shallow relations. We try to limit the annotation depth by annotating lower level relations and omiting higher level relations. As a rule of thumb, annotation depth should not be more than 2 (unless annotator is completely confident in their decision).

4.4.2 Structure

We observe that in some cases, we may have a good idea about the span of one side of a rule but not the other. For example, we may be confident of a satellite of a relation being a summary or an evaluation. However it may be more difficult to decide the span of the nucleus. We may choose to avoid creating a relation, but then we lose some information about structure. To address this, we create a DUMMY EDU and connect the satellite to it. See illustration on Figure 4.1.

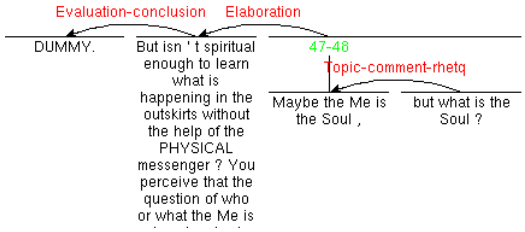

Figure 4.1: Example of RST rule with DUMMY EDU.

Bibliography

[1] A. Anderson et al. The HCRC map task corpus. *Language and Speech*, 34:351–366, 1991.

[2] H. Bunt. Dialogue pragmatics and context specification. In H. Bunt and W. Black, editors, *Abduction, Belief and Context in Dialogue: Studies in Computational Pragmatics*, volume 1 of *Natural Language Processing*, pages 81–150. John Benjamins, 2000.

[3] J. Carletta, A. Isard, and J. C. Kowtko. The reliability of a dialogue structure coding scheme. *Computational Linguistics*, 23:13–31, 1997.

[4] L. Carlson, D. Marcu, and M. E. Okurowski. Building a Discourse-Tagged Corpus in the Framework of Rhetorical Structure Theory. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*, Aalborg, Denmark, 2001.

[5] H. Clark. *Using Language.* Cambridge University Press, New York, 1996.

[6] M. Core and J. Allen. Coding Dialogs with the DAMSL Annotation Scheme. In *Working Notes: AAAI Fall Symposium on Communicative Action in Humans and Machine*, 1997.

[7] D. duVerle and H. Prendinger. A novel discourse parser based on support vector machines. In *ACL*, 2009.

[8] D. Marcu. Instructions for manually annotating the discourse structures of texts, 1999.

[9] P. Piwek. Presenting arguments as fictive dialogue. In *8th Workshop on Computational Models of Natural Argument (CMNA08; in conjunction with ECAI 2008)*, 2008.

[10] P. Piwek and K. van Deemter. Generating under Global Constraints: the Case of Scripted Dialogue. *Journal of Research on Language and Computation*, 5(2):237–263, 2007.

[11] R. Prasad et al. Penn Discourse Treebank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco, 2008.

[12] J. R. Searle. *A Taxonomy of Illocutionary Acts*. Minnesota Studies in the Philosophy of Science. University of Minnesota Press, language, mind, and knowledge edition, 1975.

[13] S. Stoyanchev and P. Piwek. Constructing the CODA corpus. In *Procs of LREC 2010*, Malta, May 2010.

[14] F. Wolf and E. Gibson. Representing discourse coherence: A corpus-based study. *Computational Linguistics*, 2(31):249–288, 2005.