Towards Automatic Dialogue Understanding

Rodolfo Delmonte, Antonella Bristot
Department of Language Science
Universita` "Ca Foscari"
30123 - VENEZIA
delmont@unive.it

Vincenzo Pallotta
Department of Computer Science
Webster University, Geneva
Switzerland
pallotta@webster.ch

Abstract

In this paper we will present work carried out to scale up the system for text understanding called GETARUNS, and port it to be used in dialogue understanding. The current goal is that of extracting automatically argumentative information in order to build argumentative structure. The long term goal is using argumentative structure to produce automatic summarization of spoken dialogues.
Very much like other deep linguistic processing systems(see Allen et al.), our system is a generic text/dialogue understanding system that can be used in connection with an ontology – WordNet - and other similar repositories of commonsense knowledge. Word sense disambiguation takes place at the level of semantic interpretation and is represented in the Discourse Model. We will present the adjustments we made in order to cope with transcribed spoken dialogues like those produced in the ICSI Berkely project. The low level component is organized according to LFG theory; at this level, the system does pronominal binding, quantifier raising and temporal interpretation. The high level component is where the Discourse Model is created from the Logical Form. For longer sentences the system switches from the topdown to the bottomup system. In case of failure it will backoff to the partial system which produces a very lean and shallow semantics with no inference rules.
In a final section we present preliminary evaluation of the system on two tasks: the task of automatic argumentative labeling and another frequently addressed task: referential vs. non-referential pronominal detection. Results obtained fair much higher than those reported in similar experiments with machine learning approaches.

1. Introduction

The system presented here has been achieved in over two decades with the goal of developing a broad-coverage, domain general natural language unterstanding system. The underlying grammar, lexicon, the semantics and all intermediate modules are intended to be domain-general and to be easily portable to different application domains. As is the case with all rule-based systems, (but see also Allen et al., 2007), we have no need to collect and annotate corpora for specific subtasks because the system already has good performance in all current parsing and semantic related tasks (see Delmonte et al. 2006; Delmonte 2007 and 2008).
However, when we started last year to use the system to parse ICSI dialogues, we realized that the semantic representation and the output of the parser were both inadequate. So we worked at deficiencies we detected in an empirical manner. This approach made us aware of the peculiarities of spoken dialogue texts such as the ones made available in ICSI project (see Janin et al. 2003), and to the way to implement solutions in such a complex system. These dialogues are characterized by the need to argument in a exhaustive manner the topics to be debated which are the theme of each multiparty dialogue. The mean length of utterances/turns in each dialogue we parsed was rather long. This makes ICSI dialogues hard to compute. From a count of number of words x turn, we came up with the following average figures:
- percent of turns made of one single word: 30%
- percent of turns made of up to three words: 40%
- number of words x turn overall: 7

- number of words x turn after subtracting short utterances: 11
These values correspond to those found for PennTreebank corpus where we can count up to 94K sentences for 1M words – again 11 words per sentence. In analyzing ICSI, we found turns with as much as 54 words depending on the topic under discussion and on the people on the floor.
Computing semantic representations for spoken dialogues is a particularly hard task which – when compared to written text processing - requires the following additional information to be made available:
- adequate treatment of fragments;
- adequate treatment of short turns, in particular one/two-words turns;
- adequate treatment of first person singular and plural pronominal expressions;
- adequate treatment of disfluencies, thus including cases of turns made up of just such expressions, or cases when they are found inside the utterance;
- adequate treatment of overlaps;
- adequate treatment of speaker identity for pronominal coreference;
In addition, in our system every dialogue turn receives one polarity label, indicating negativity or positivity, and this is computed by looking into a dictionary of polarity items. This is subsequently used to decide on argumentative automatic classification.
We will address each such topics in what follows. The paper is organized with a presentation of the system in section 2; then problems will be presented and discussed by addressing their computational role and the module responsible for its treatment, in the following section. We will then present some data and an evaluation. A final section will be devoted to conclusions.

2. The System GETARUNS

GETARUNS[1], the system for text understanding developed at the University of Venice, is organized as a pipeline which includes two versions of the system: what we call the Partial and the Deep GETARUNS. At first we will present the Deep version, which is equipped with three main modules: a lower module for parsing, where sentence strategies are implemented; a middle module for semantic interpretation and discourse model construction which is cast into Situation Semantics; and a higher module where reasoning and generation takes place.

The system is based on LFG theoretical framework (see Bresnan, 2000) and has a highly interconnected modular structure. The Closed Domain version of the system is a top-down depth-first DCG-based parser written in Prolog Horn Clauses, which uses a strong deterministic policy by means of a lookahead mechanism with a WFST to help recovery when failure is unavoidable due to strong attachment ambiguity.

It is divided up into a pipeline of sequential but independent modules which realize the subdivision of a parsing scheme as proposed in LFG theory where a c-structure is built before the f-structure can be projected by unification into a DAG (Direct Acyclic Graph). In this sense we try to apply in a given sequence phrase-structure rules as they are ordered in the grammar: whenever a syntactic constituent is successfully built, it is checked for semantic consistency. In case the governing predicate expects obligatory arguments to be lexically realized they will be searched and checked for uniqueness and coherence as LFG grammaticality principles require.

Syntactic and semantic information is accessed and used as soon as possible: in particular, both categorial and subcategorization information attached to predicates in the lexicon is extracted as

[1] The system has been tested in STEP competition (see Delmonte 2008), and can be downloaded at, http://project.cgm.unive.it/html/sharedtask/.

soon as the main predicate is processed, be it adjective, noun or verb, and is used to subsequently restrict the number of possible structures to be built. Adjuncts are computed by semantic compatibility tests on the basis of selectional restrictions of main predicates and adjuncts heads.

The output of grammatical modules is fed then onto the Binding Module which activates an algorithm for anaphoric binding. Antecedents for pronouns are ranked according to grammatical function, semantic role, inherent features and their position at f-structure. Eventually, this information is added into the original f-structure graph and then passed on to the Discourse Module (hence DM).

The grammar is equipped with a core lexicon containing most frequent 5000 fully specified inflected word forms where each entry is followed by its lemma and a list of morphological features, organised in the form of attribute-value pairs. However, a morphological analyser for English is also available with big root dictionaries (25,000 for English) which only provide for syntactic subcategorization, though. In addition to that, there are all lexical form provided by a fully revised version of COMLEX, and in order to take into account phrasal and adverbial verbal compound forms, we also use lexical entries made available by UPenn and TAG encoding. Their grammatical verbal syntactic codes have then been adapted to our formalism and are used to generate a subcategorization scheme with an aspectual and semantic class associated to it – however no restrictions can reasonably be formulated on arguments of predicates. Semantic inherent features for Out of Vocabulary Words, be they nouns, verbs, adjectives or adverbs, are provided by a fully revised version of WordNet - plus EuroWordnet, with a number of additions coming from computer, economics, and advertising semantic fields - in which we used 75 semantic classes similar to those provided by CoreLex. The complete lexicon counts 270K entries.

When each sentence is parsed, tense aspect and temporal adjuncts are accessed to build the basic temporal interpretation to be used by the temporal reasoner. Eventually two important modules are fired: Quantifier Raising and Pronominal Binding. QR is computed on f-structure which is represented internally as a DAG. It may introduce a pair of functional components: an operator in the structure where the quantifier can be raised, and a pool containing the associated variable where the quantifier is actually placed in the f-structure representation. This information may then be used by the following higher system to inspect quantifier scope. Pronominal binding is carried out at first at sentence internal level. DAGs will be searched for binding domains and antecedents matched to the pronouns if any to produce a list of possible bindings. Best candidates will then be chosen.

2.1 The Upper Module

GETARUNS, has a highly sophisticated linguistically based semantic module which is used to build up the DM. Semantic processing is strongly modularized and distributed amongst a number of different submodules which take care of Spatio-Temporal Reasoning, Discourse Level Anaphora Resolution, and other subsidiary processes like Topic Hierarchy which cooperate to find the most probable antecedent of coreferring and cospecifying referential expressions when creating semantic individuals. These are then asserted in the DM, which is then the sole knowledge representation used to solve nominal coreference, before proceeding to access external knowledge in the ontologies. The system uses two resolution submodules which work in sequence: they constitute independent modules and allow no backtracking. The first one is fired whenever a free sentence external pronoun is spotted; the second one takes the results of the first submodule and checks for nominal anaphora. They have access to all data structures contemporarily and pass the resolved pair, anaphor-antecedent to the following modules. Semantic Mapping is performed in two steps: at first a Logical Form is produced which is a structural mapping from DAGs onto unscoped well-formed formulas. These are then turned into situational semantics informational units, infons which may become facts or "sits". Each unit has a relation, a list of arguments which in our case receive their semantic roles from lower processing – a polarity, a temporal and a spatial location index. The clause-level interpretation procedure interprets clauses on the basis of lexical properties of the governing verb. This is often non available in short turns and in fragments. So in many cases, fragments are built into a sentence by inserting a dummy verb which varies from dummy BE or dummy SAY depending on speech act present.

3 The Spoken Dialogue Additions

We will proceed by addressing each problem presented above in the order with which it is coped with by the system.

3.1 The Algorithm for Overlaps

Overlaps are an important component of all spoken dialogue analysis (Delmonte 2003). In all dialogue transcriptions, overlaps are treated as a separate turn from the one in which they occur, which usually follows it. This is clearly wrong from a computational point of view. For this reason, when computing overlaps we set as our first goal that of recovering the temporal order in which speaker and interlocutor interact. This is done because:

- overlaps may introduce linguistic elements which influence the local context;
- eventually, they may determine the interpretation of the current utterance.

For these reasons, they cannot be moved to a separate turn because they must be semantically interpreted where they temporally belong. In addition, overlaps are very frequent. The algorithm we built looks at time stamps, and everytime the following turn begins at a time preceding the ending time of current turn it enters a special recursive procedure. It looks for internal interruption in the current turn and splits the utterance where the interruption occurs. Then it parses the split initial portion of current utterance and continues with the overlapping turn. This may be reiterated in case another overlap follows which again begins before the end of current utterance. Eventually, it returns to the analysis of the current turn with the remaining portion of current utterance.

In Table 1 below we present data related to overlaps for the first 10 dialogues we computed. We classified overlaps into two types – WHILE and AFTER - according to whether they take place inside the turn of the current speaker or at the end. The second case being regarded as normal and non disrupting of the current speaker's conversational plan.

	total	Cont-inue	Interr-upt	inter_cont	inter_change	inter_other
turns	13158	-	-	-	-	-
while	1624	1369	46	87	22	63
after	1461	-	-	-	-	-

Table 1. Overlaps and their effects on Planning

On a total number of 13158 turns we thus computed 3085 overlaps divided up nicely almost half and half for each of the two classes. Then we proceeded by subdividing WHILE overlaps into 5 subclasses where Continue indicates the current speaker continues talking; Interrupt, the current speaker is interrupted and there is no continuation; Inter_Cont, the current speaker is interrupted but then Continues his/her plan in a following turn; Inter_Change, the current speaker is interrupted and changes his/her plan, bu either changing subject topic, or answering the overlapper. Eventually we had Inter_Other which indicates cases in which dialogue is interrupted by other speakers.
As can be easily noticed, the case constituted by Inter_Change which is the most interesting from a semantic and pragmatica point of view is in fact the less frequent. We assume, however, that this may be determined by other factors attaining to the type of conversation being entertained by the participants, as well as by the nature of the topics discussed, and eventually by the personalities of the interlocutors.

3.2 The Treatment of Fragments and Short Turns

Fragments and short turns are filtered by a lexical lookup procedure that searches for specific linguistic elements which are part of a list of backchannels, acknowledgements expressions and other similar speech acts. In case this procedure has success, no further computation takes place. However, this only applies to utterances shorter than 5 words, and should be made up only of such special words. No other linguistic element should be present apart from non-words, that is words which are only partially produced and have been transcribed with a dash at the end.
- graceful failure procedures for ungrammatical sentences, which might be fullfledged utterances but semantically uninterpretable due to the presence of repetitions, false starts and similar disfluency phenomena. Or else they may be just fragments, i.e. partial or incomplete utterances, hence non-interpretable as such; this is done by imposing grammatical constraints of wellformedness in the parser;
- failure procedures for utterances which are constituted just by disfluency items and no linguistically interpretable words. These must be treated as semantically empty utterances and are recognizable by the presence of orthographic signs indicating that the word/s have not been completed and are just incomprehensible; this is done by inspecting the input in search of special orthographic marks and preventing the utterance to be passed down to the partial/deep parser.
On the contrary, we implemented a principled treatment of elliptical utterances which contribute one specific speech act or communicative act. They may express agreement/disagreement, acknowledgements, assessments, continuers etc. All these items are computed as being complements of abstract verb SAY which is introduced in the analysis, and has as subject, the name of current speaker.

4 Automatic Argumentative Annotation

At first we shall provide a state of the art and then we shall comment in detail our approach.

4.1 Detecting Argumentative structure – issues and theories

As shown by Rosemberg and Silince (1999), tracking argumentative information from meeting discussions is of central importance for building summaries of project memories since, in addition to the "strictly factual, technical information", these memories must also store relevant information about decision-making processes. In a business context, the information derived

from meetings is useful for future business processes, as it can explain phenomena and past decisions and can support future actions by mining and assessment (Pallotta et al., 2004).
In a section below we will descrive in detail how the annotation process takes place. Here we want to highlight the main features of this process. This first level of annotation is based on the shallow dialogue model, proposed in (Armstrong, 2003), of which it is a modified version. This model provides a simple operational structure of dialogues based on three categories:

• ***a dialog*** is a non empty set of episodes; a new episode is identified by a topic/speaker shift.
• ***an episode*** is a non empty set of turns; turns are individuated at prosodic level – more on turns below.
• ***a turn*** is a non empty sequence of clauses/utterances and their boundary is a long pause.

In addition to the shallow dialogue model, we consider the adoption of a deeper structured representation based on argumentation theory. We assume that meeting dialogues are better viewed from the Collaborative Decision Making (CDM) perspective. In CDM, a meeting is defined as a multi-party (multi-agent) decision making process: a collaborative process, where agents follow a series of communicative actions in order to establish a common ground on the dimension of the problem. The main four dimensions of CDM process are:

• an overall task issue;
• a set of alternative proposals;
• a set of arguments in favor or against each proposals;
• a collection of choice criteria (perspectives and preferences) settled upon the participants;
• a decision (or evaluation) function that combines criteria to judge the alternatives.

This definition focuses on the processes, which take place during meetings and how these processes contribute to the accomplishment of a joint goal. In order to capture the above dimensions, we then adopted and extended a suitable argumentative model of discussions, namely the IBIS model proposed by (Kunz and Rittel, 1970). The IBIS model provides us with an abstract description of the discussion's rationale by outlining the important points discussed, the conflicts arisen and, hopefully solved, and the decisions that have been made. The IBIS model abstracts from the dynamics of the discussion, which needs to be modeled as well in order to extract the IBIS structures from meeting events. Relevant meeting events are special types of Dialogue Acts that have an argumentative force. This type of Dialogue Acts (Bunt 1979) called Argumentative Acts, are backward-looking acts with forward-looking expectations (Goffman 1981).
Within the Adjacency Pairs model (Schegloff & Sacks 1973), the importance of tracking agreement and disagreement in discussions has been recognized also in (Galley et al., 2004; Hillard, Ostendorf, and Shriberg, 2003). Although these methods have the great advantage of being automatic, they only partially help in reconstructing the argumentative information we need in order to answer real user queries. This model has been adopted by (Niekrasz et al. 2005) for the real-time reconstruction of an argumentative structure by overhearing discussions in design meetings. Finally, (Rienks and Verbree 2006) propose the Twente Annotation Schema that is based on fewer categories but more relation types being inspired by the Rhetorical Structure Theory (Mann and Thompson 1988).
The argumentative structure defines the different patterns of argumentation used by participants in the dialogue, as well as their organization and synchronization in the discussion. The limits of sequential analysis of conversation (Schegloff & Sacks 1973) have been already pointed out by (Goffman 1981), who proposes to extend the notion of adjacency pair with that of chains of interaction rounds. As for other related work, we also see similarities of our approach with the argumentation dependency grammar proposed by (Lo Cascio 1991), although in his work only

argumentative structure of monologues is considered. In fact, when analyzing dialogues, adjacency pairs are not enough to represent the hierarchical structure of the discussion: consider, for instance, an answer that replies to two different questions in the discussion. In this case, we need to add a relation that links the answer to both of the questions. We call this relation "replies_to". The “replies_to” links a (re)action to one or more previous (possibly in time) actions and induces an argumentative chain structure on the dialogue, which is local to each action and which enables the visualization of its context. For instance, the context of the action of “accepting a clarification” will be a chain of linked actions, namely the action of the clarification, that of the proposal that is clarified and the action of raising an issue for which the proposal was made. Argumentative actions can overlap in time, as for instance in those cases where the acceptance of a justification is uttered in the form of “backchannel” during the presentation of the justification.

Argumentative actions such as REQUEST, ACCEPT, REJECT might correspond to basic dialogue acts (Clark and Popescu-Belis 2004). In this case we have refined the concept of dialogue act and adjacency pairs by specifying the role of dialogue acts in constructing the argumentative structure of the discussion through the “replies_to” relation.

When using the IBIS mark-up labels, a meeting is decomposed into several stages such as issues, proposals, and positions, each stage being possibly related to specific aggregations of elementary dialogue acts. Moreover, argumentative interactions may be viewed as specific parts of the discussion where several dialogue acts are combined to build such an interaction; as for instance, a disagreement could be seen as an aggregation of several acts of reject and accept of the same proposal. From this perspective, we elaborated an argumentative coding scheme, the Meeting Description Schema (Pallotta et al. 2004), which takes into account the different stages (or episodes) defined by the IBIS model and extend the concept of adjacency pairs to relate these episodes to each other and to the corresponding argumentative function.

In MDS, the argumentative structure of a meeting is composed of a set of topic discussion episodes (a discussion about a specific topic). In each discussing topic, there exists a set of issue discussion episodes. An issue is generally a local problem in a larger topic to be discussed and solved. Participants propose alternatives, solutions, opinions, ideas, etc. in order to achieve a satisfactory decision. Meanwhile, participants either express their positions and standpoints through acts of accepting or rejecting proposals, or by asking questions related to the current proposals. Hence, for each issue, there is a corresponding set of proposals episodes (solutions, alternatives, ideas, etc.) that are linked to a certain number of related positions episodes (for example a rejection to a proposed alternative in a discussing issue) or questions and answers.

4.2 Our Approach

Automatic Argumentative Annotation, is carried out by a special module activated at the very end of the computation of the each dialogue. This module takes as input the complete semantic representation produced by the system recorded in Prolog facts in the DM. The elements of semantic representation we use are the following ones:

- all facts in Situation Semantics contained in the Discourse Model, which include individuals, sets, classes, cardinality, properties related to entities by means of their semantic indices;
- facts related to spatiotemporal locations of events with logical operators and semantic indices;
- vectors of informational structure containing semantic information at propositional level, computed for each clause;

- vectors of discourse structure with discourse relations computed for each clause from informational structure and previous discourse state (for an evaluation of system's performance see Delmonte et al. 2007);
- dialogue acts labels associated to each utterance or turn following ICSI classification;
- overlaps information computed at utterance level;
- topic labels associated to semantic indices of each entity marked as topic of discourse;
- all utterances with their indices as they have been automatically split by the system.

To produce Argumentative annotation, the system uses the following 21 Discourse Relations labels:

statement, narration, adverse, result, cause, motivation, explanation, question, hypothesis, elaboration, permission, inception, circumstance, obligation, evaluation, agreement, contrast, evidence, hypoth, setting, prohibition

These are then mapped onto five general argumentative labels. In addition we use the label DISFLUENCY for all those turns that contain fragments which are non-sentences and are semantically uninterpretable.

ACCEPT,
REJECT/DISAGREE,
PROPOSE/SUGGEST,
EXPLAIN/JUSTIFY,
REQUEST
DISFLUENCY

The algorithm works in the following manner:
1. It recovers Dialogue Acts for each dialogue turn as they have been assigned by the system. These labels coincide with ICSI labels (BKC, ACK, FGB, FHD, RHQ, - that is Floor Grabber, Floor Holder, Backchannel, Acknowledge, RhetoricQuestion - with the addition of NEGation, ASSent, MTVation, PRPosal, GRTeeing, CNLusion;
2. It recovers Overlaps as they have been marked during the analysis;
3. It produces an Opinion label which we call Polarity, which can take one of two values: Positive or Negative according to whether the sentence contains positive or negative linguistic descriptions;
4. It produces a list of Hot Spots and builds up Episodes, where Hot Spots is simply a set of turns in sequence where the interlocutors overlap each other frequently. Episodes on the contrary are a set of turns in which a single speaker "arguments" his/her topics which may occasionally be interrupted by overlaps or by short continuers, backchannel or other similar phenomena by other speakers without however grabbing the floor;
5. Then the main predicate that assigns argumentative labels is called:
   i. at first it tries exceptions on the basis of the actual words contained in the turn. These exceptions may be constituted by Greetings, specific Speech Acts, Conventional utterances pronounced in specific situations like Thanking, etc.;
   ii. then Short utterances are checked. In case they end up with a question mark they are labeled as Questions. Else, the Dialogue Act label is considered. Negations are also computed here;
   iii. now the main call is activated. In order to start matching the rules, the semantic information is recovered for the current turn, clause by clause;

iv. when semantic information has been recovered the rules are fired. There are some 33 rules which take as input the following vector of features:

assignargument(NoCl, [Pol,DialAct], DiscDom, DiscRel, Relev, DomPointView, Output) at

where Output is the output label chosen by the rule; DiscDom may be Factive or NonFactive, Suggestion or Proposal; Relevance may be foreground or background; DomPointView may be objective or subjective. Rulea are applied by matching input labels in a FSA manner. However sometimes conditions and constraints are made to apply. For instance, ***analyzecontext(NoCl)***, checks to verify whether the current speaker holds the floor in the 2 preceding or following clauses.

v. the rules produce a set of argumentative labels, one for each clause. The system then chooses the label to associate to the turn utterance from a hierarchy of argumentative labels graded for Pragmatic Relevance which establishes that, for instance, Question is more relevant than Negation, which is more relevant than Raise Issue, etc.

A short example extracted from Dialogue 1 is reported in the Appendix at the end of the paper where we report for each utterance the Predicate-Argument structure and the governing predicate/s preceded by the argumentative label chosen by the system. Eventually we are able to evaluate the degree of collaboration vs. competitiveness of each participant in the conversation and make a general statement like this one produced automatically for Dialogue 1,

The speaker that has held the majority of turns is
- Don with a total of 512 turns,
followed by
- Morgan with a total of 456;
The speaker that has undergone the majority of overlaps is Morgan followed by Jane;
The speaker that has done the majority of overlaps is Morgan followed by Jane;
Morgan is the participant that has been most competitive.

The system has been used to parse the first 10 dialogues of the ICSI corpus for a total number of 98523 words and 13803 turns. This has been done to "train" the system: what happened was that, for the first 5 dialogues, we had to take care of failures. We also had to tune all the modules and procedures carefully. In particular, the module for argumentative automatic classification was incrementally improved in order to cover all conventional ways to express Agreement. For this reason, we then chose two random additional dialogues to test this second task.

4.3 Experimental Results

We had one skilled linguist to provide a turn level annotation for argumentative labels: we don't have any agreement measure in this case, even though we expect the annotation to be in line with current experiments on the same subject (Pallotta et al. 2007). In the following table we report data related to the experiment of automatic annotation of argumentative categories. On a total of 2304 turns, 2251 have received an argumentative automatic classification, with a Recall of 97.53%. As can be gathered from the following table 2., the F-score is fairly high compared to current results reported in the literature on the same topic which are all below 80%.

	Correct	Incorrect	Total Found
Accept	662	16	678
Reject	64	18	82

Propose	321	74	395
Request	180	1	181
Explain	580	312	892
Disfluency	19		19
Total	1826	421	2247

Table 2. Overall count of argumentative labels

We computed Precision as the ratio between Correct Argumentative Labels/Found Argumentative Labels, which corresponds to 81.26%. The F-score is 88.65%.

5. The Anaphora Resolution Module

The problem represented by pronominal expressions in dialogues needs to be addressed fully and not by means of ad hoc solutions. This requires a full-fledged system for anaphora resolution. One such system is shown in Fig. 1 below, where we highlight the architecture and main processes undergoing at the anaphora level. First of all, the subdivision of the system into two levels: Clause level – intrasentential pronominal phenomena – where all pronominal expressions contained in modifiers, adjuncts or complement clauses receive their antecedent locally. Possessive pronouns, pronouns contained in relative clauses and complement clauses choose preferentially their antecedents from list of higher level referring expressions. Not so for those pronouns contained in matrix clauses. In particular the ones in subject position are to be coreferred in the discourse. This requires the system to be equipped with a History List of all referring expressions to be used when needed. In the system, three levels are indicated: Clause level, i.e. simple sentences; Utterance level, i.e. complex sentences; Discourse level, i.e. intersententially.

Our system computes semantic structures in a sentence by sentence fashion and any information useful to carry out anaphoric processes needs to be made available to the following stretch of dialogue.

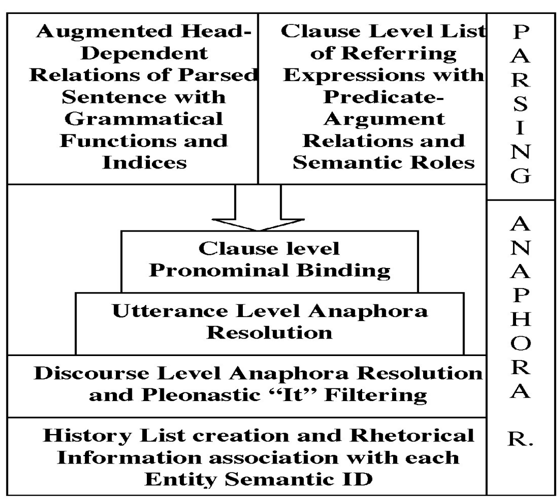

Figure 1. Anaphoric Processes in GETARUNS

6. The Experiments

We set up a number of experiments in order to test the new version of the system on the task of detecting referential from nonreferential uses of personal pronouns YOU, WE and the pronoun IT.

6.1 State of the Art

Although much has been written about the referentiality/nonreferentiality of pronouns in written text (mostly the third person neutral pronoun *it*), only recently the interest has shifted on spoken dialogues, partly thanks to the availability of annotated corpora. The main problems when trying to work with spontaneous speech are constituted by the presence of disfluencies, hesitations, abandoned utterances, interruptions, backchannels, etc.
According to Strube and Müller, another problem is represented by the fact that spontaneous speech contains more instances of referential and nonreferential pronouns than written text and also that spontanuous speech is characterized by the presence of a large number of singletons among the pronouns that are usually expletive or *vague* and cause problems for pronoun resolution algorithms which usually attempt to always find an antecedent.

6.1.1 You. Very few studies deal with the analysis on the referential/nonreferential use of *you* in spoken dialogue. Y*ou* is an ambiguous pronoun because it can either be generic, i.e. does not refer to an addressee or to one of the participants in the dialogue, or referential and in this case, if singular, refers to the addressee of current utterance; else in case it is plural, it has more than one antecedent in previous discourse stretch. In their experiments for the resolution of *you* based on a portion of the Switchboard Corpus consisting of two-party telephone conversations, Gupta et al.(2007) distinguish between the nonreferential and referential uses of *you* and also provide a few hints at the identification and classification of some idiomatic expressions, such as *you know* or clusters like *two of you, some of you,* etc, where *you* is always generic. Their analysis, primarily based on two-person conversations, has recently been applied quite successfully to multi-party dialogues.
According to the authors, in multi-party dialogues it is very important to identify the speaker and determine whether he is directly referred to using a second person pronoun or not. In spoken dialogue, in fact, most cases of *you* seem to be generic. They identify three types of *you*: generic, referential and reported referential (when *you* is mentioned in reported speech). They extract a number of features (Sentential, POS, Dialogue Act and Oracle Context features) directly derived from the corpus; the context features carry information (correct generic/referential label) about the uses of *you* in preceding context. Also some other features that may be of importance in detecting the correct role of *you* in dialogues, i.e. the presence of a question mark, are taken into consideration.
Even though the overall performance of the system is quite good with high precision values, 94% for the highest-accuracy result with only 36 errors out of 673 cases found, and accuracy by using all sets of features reaching a value of 84.4%, the authors realize how the use of some features, especially the context feature (accuracy=45.66%), are ininfluent in the analysis, while others perform quite well (dialogue act=80.92%). This is motivated by the fact that the generic/referential status of *you* may be independent of previous *yous* and that only looking at the neighbouring context may not be sufficient for a correct analysis of this pronoun. They also checked some prosodic features which apparently did not add useful information.
In a second analysis performed on a 15-meeting subset of the AMI Meeting Corpus the classification of *you* resulting from the annotation of the spoken text is of four kinds: generic,

referential, reported speech and discourse marker (used to mark cases of *you* in idiomatic expressions and clusters as mentioned above). The authors decided to focus mainly on the categories generic/referential because the others occur in less than 2% of the dataset. For multi-party dialogues the result are worse than the two-speaker ones, resulting in a value of accuracy around 75 % when using all the features derived from their previous work. They realize how multi-party dialogues are much more complex than two-party dialogue. A further analysis is then made only on the cases of referential *you* with the aim of identifying the addressee; this is achieved through a four-value model where value 1 is given to the potential addressee and value 4 addressing the entire group of speakers. The analysis is based on three sets of features (structural, durational and lexical) and the system reaches an accuracy of 47.2%.
Yavanovich and op den Akker in their study for the addressee detection in face to face meetings of the MRDA Corpus, notice how *you* may often refer to meeting participants excluding the speaker of current utterance. The usage of quantifying determiners, numerals and infinitive pronouns may help in distinguishing *you* as a specific person from *you* as a group. At the same time, first name detection can be a very important means for addressee determination, especially when used in the vocative form. They also noticed that the speaker of utterances marked with question tags usually speaks directly to the addressee to provide information. This may be very important whenever there is a need to identify the addressee.
The method illustrated by Javanovich and al. seems to be quite good in determining addresse identification reaching an accuracy of 83.74% also because they use information derived from the study of some important non linguistic features such as gaze direction, gesture and context.

6.1.2 We. To our knowledge studies on the referential/non referential nature of the pronoun *we* in spoken text are rare. Javanovich and op den Akker [4] merely mention *we* observing how in multi-party dialogues *we* can either be used to refer to a subgroup of people taking part to the meeting or to all the participants, rather than to a single person. This is quite evident in open-ended questions that do not require a specific answer and which are usually addressed to all meeting partecipants; while this is not true for *you,* whenever an open question contains this pronoun it is usually referred to a single speaker.

6.1.3 It. Several studies deal with the automatic classification of this pronoun, even though only a very few of them focus on the role of *it* in spoken dialogues. The problem with the pronoun *it* is two-fold: *it* can be generic, referential and expletive; morover, *it* can have NP- and non-NP-antecedents, especially in spoken dialogues.
In their study Strube and Müller show how *it* can be classified as referential or nonreferential by taking into consideration the local context surrounding the pronoun. Spoken text contains more pronouns with non-NP-antecedents than written text; specifically, *it* is a *vague* pronoun that may refer to different kinds of abstract objects from previous discourse: sentential antecedents or VP-antecedents. In their work, this class of vague pronouns also includes expletive pronouns, which are nonreferential at all, and the so-called *vague anaphors*, for which humans don't seem to be able to determine an antecedent.
Their analysis, operated by a system originally used on written text and extended with a set of features designed for spoken dialogue, is based on part of the Switchboard Corpus that consists of 16.601 markables (sequence of words and attributes associated with them). Most of the pronouns found in the corpus are singletons which are vague or expletive and cause major problems in the processing of reference resolution algorithms. Strube and Müller's recognition is based on two classes of features used for written textxs: NP-level (grammar level) and Coreference-level

features to which a new group of feature for the analysis of spoken dialogues has been added which include the type of antedecent (NP, VP, S) or the preference that some verbs exhibit for arguments of different types. The study is conducted on three sets of third person pronouns: feminine and masculine, plural and neuter with a special interest in the latter since the neuter pronouns may have non-NP-antecedents. The results are quite good for the feminine, masculine and plural pronouns, but quite low for the neuter pronoun (~40.00% precision/~8.72% recall/~14.68% f-measure) and this is mostly due to the presence of many singletons and vague pronouns which do not have an antecedent and for which their model finds one nonetheless. The values of the analysis on a total of 1250 pronouns are: precision=56.74, recall=40.72, f-measure=47.82.

In a more recent work, Müller (2006) presents an implemented version of the system which was developed on the basis of meeting dialogues of the ICSI corpus. The task described is to identify and filter out nonreferential cases of *it*; in this new version of the system there's no a priori knowledge available as to wether *it* is referential or not. According to the author, the best approach for the recognition of nonreferential *it* should be based on filtering conditions and not solely on selection conditions.

The instances of *it* in five of the dialogues of the ICSI Corpus have been manually annotated by two human annotators and classified as belonging to six different classes : *normal*, *vague* – two sub-types of referential *it* - and *discarded*, *extrapos it*, *prop-it* and *other* – used to define cases of nonreferential *it*. After this annotation *K scores* have been calculated in order to check the reliability of the annotation. On a total of 1.017 examples of *it* 62.5% are referential. After some editing and preprocessing the dialogues have been classified according to a vector of features: some represent the syntactic patterns of the text capturing the left and right context of *it*, others contain lexical information about the predicative content of *it*, a third group of features captures a wider context of *it*, finally a fourth group contains some features used in order to check wether the pronoun is preceded by a preposition or if *it* appears as subject of some verbs like *seem*, *look*, *appear*, etc. In a second step machine learning techniques have been applied to detect nonreferential *it.* The good results obtained by using information about interrupting points and sentence boundaries (precision:80.0%, recall:60.9%, f-score:69.2) is, according to the author, a sign of the utility of classyfing *it* as nonreferential in multi-party dialogues. Also the lower results obtained by simply employing automatically determined features show how the use of filtering procedures for nonreferential *it* represents an easy way to deal with the problem.

In their recent work Bergsma et al. focus on *it* because, as they point out "*it* is one of the most frequent words in the English language, accounting for about 1% of tokens in text and over a quarter of all third-person pronouns[2]. Usually between a quarter and a half of *it* instances are non referential". Although not tested on spoken text, they used the Google Web 1T 5-gram Corpus, they present an analysis that may provide some good hints at the classification of *it* in spoken language. Their system, COMBO, is based on methods that do not rely heavily on hand-crafting of specific features like many other systems tend to do. COMBO deals mainly with the extraction of *context pattens* where some *pattern fillers* (i.e. words that can take the place of the pronoun) are extracted from a large number of n-grams (to a maximum of 5-grams) in order to determine which *it* can be replaced by an NP and which cannot; to identify nonreferential *it* they count how often the pronoun appears as a pattern filler. They show how the position of *it* in the context pattern is usually fundamental in determining the referential/ nonreferential nature of *it*. Compared to other systems COMBO performs quite well as for F-score values. Unfortunately the

[2] E.g. http://ucrel.lancs.ac.uk/bncfreq/flists.html

system seems to detect only pronouns that are referential to a sentence and not to a noun - hence the necessity to extend the analysis to preceding discourse - and, above all, if the position of the pattern filler cannot be occupied for various reasons a possible referential occurrence of the pronoun is lost.

7. The Experiment: Method and Discussion

This work has tried to solve problems related to referential vs generic interpretation of pronominal expressions "it", "you" and "we".
In order to take decisions as to whether pronouns are to be interpreted as referential or not a recursive procedure checks the type of governing predicate. Referential pronouns are then passed on to the pronominal binding algorithm that looks for local antecedents if any. Otherwise, the pronouns is labeled as having External coreference in the previous discourse stretch. The Anaphora Resolution module will then take care of the antecedent and a suitable semantic identifier will be associated to it. On the contrary, if the pronouns are judged to be referentially empty or generic, no binding takes place. The pronoun is associated to the label "arbitrary" which prevents the system from computing it as a referential expression in the following semantic interpretation module. Not being a semantically interpretable linguistic element is important in order to avoid the pronoun from entering the Topic Hierarchy Centering module and being assigned a score. At the same time, the pronominal linguistic expression has to be used in order to complete predicate-argument structure. We assume that using such referentially empty pronouns corresponds to using expletive "it" in agentless passive constructions, or expletive “there” in presentative sentences.
To satisfy these constraints we let the parser compute these generic pronouns as PREDs and be associated to an f-structure. Then the pronominal binding module associates the label "arbitrary" to the value INTERPRETATION (it can be quantified or definite), and this labels is then used by the Anaphora Resolution Module to discard the pronoun from the list of possible referring expressions.
We intended to evaluate the system at different levels of computation so we organized an experiment related to a classical task in spoken dialogue system evaluation, distinguishing referring from non-referring pronominal expressions. This task was targeted to the sentence level processing modules: in particular, the “Pronominal Binding Module”(hence PBM) which is responsible for the search of Antecedents of pronouns in the current utterance, is fired after Quantifier Raising has been attempted. The output of the PBM is added to the DAG of current structural representation and is used by the Discourse Level processing modules to execute anaphora resolution at Discourse level – for an evaluation of system’s performance see Delmonte et al. 2006. In fact, what we did was to use the same setup we already organized for written text in relation to the pronoun IT, without any additional heuristics. We extended the search for “generic” non-referential pronouns to WE and YOU.

7.1 Experimental Results

The system has been used to parse the first 10 dialogues of the ICSI corpus for a total number of 98523 words and 13803 turns. This has been done to “train” the system: what happened was that, for the first 5 dialogues, we had to take care of failures. We also had to tune all the modules and procedures carefully.

In the experiment the system has to make a two-way decision and choose to evaluate pronouns either as referential or non-referential. Here below is a table containing total values for pronouns WE/YOU/IT in all the 10 dialogues analysed.

	Referent	Generic	Total	Found
WE	1186	706	1892	1356
YOU	1045	742	1787	1132
IT	1593	1008	2601	1627
Total	3824	2456	6280	4115

Table 3. Overall count of pronominal expressions

We had two skilled linguists to annotate pronominal WE/IT/YOU properties as either referential/ nonreferential. Their agreement on this task was very high with a kappa-score of 0.71. Results for the experiment are as follows,

	Recall	Precision	F-Score
WE	71.67%	81.2%	76.14%
YOU	63.34%	89.3%	74.11%
IT	62.52%	84.6%	72.19%

Table 4. Results for pronominal expressions

8 Conclusions and Future Work

We have presented work carried out to extend and adapt a system for text understanding in order to make it fit for dialogue understanding. We proposed a set of expansions to cope with typical dialogue related problems, such as presence of non-sentential fragments, elliptical fragments interpretable as speech acts, massive presence of generic non-referential pronominal expressions, etc. We implemented a number of additional components: an algorithm that takes care of overlaps and uses that information to split current utterances and temporally realign the conversational flow. A module that computes Argumentative automatic classification labels out of a small set, on top of discourse relations and other semantic markers determined by the semantic component of the system.

The system has been evaluated for two of its most important components, the newly implemented pronominal binding module and the argumentative classification module. Results are very encouraging. However, we note that in that task, labels which may cause great uncertainty and are highly ambiguous, have been lumped together to facilitate the classification task.

Of course we intend to complete the analysis of all dialogues contained in the ICSI corpus and refine our algorithms. Then we would like to use the system with a totally different scenario, as for instance the Switchboard two parties dialogues and see whether the “training” carried out on the basis of multiparty dialogues may be fruitfully applied to such reduced conversational framework. In particular we still need to work at the level of DECISION labeling, and to improve the discrimination of really argumentative from pragmatically irrelevant utterance, a choice that in some cases is hard to make on an automatic basis.

References

Allen, J., M. Dzikovska, M. Manshadi, and M. Swift. 2007. Deep linguistic processing for spoken dialogue systems. In ACL 2007 Workshop on Deep Linguistic Processing, pp. 49–56.

James F. Allen, Mary Swift, Will de Beaumont, 2008. Deep Semantic Analysis of Text, in Proc.STEP08, Venice, pp. 343-354.

Armstrong, S. et al. 2003. Natural language queries on natural language data: a database of meeting dialogues. In Proceedings of NLDB’2003 conference, Burg/Cottbus, Germany.

Bunt, H. Conversational principles in question-answer dialogues. In D. Krallmann, editor, Zur Theory der Frage, pages 119–141. Narr Verlag, Tübingen, 1979.

Shane Bergsma, Dekang Lin and Randy Goebel. 2008. Distributional Identification of Non-Referential Pronouns, In ACL-HLT 2008, Columbus, Ohio, June 16-18, 2008, pp. 10-18.

Adriane Boyd, Whitney Gegg-Harrison, and Donna Byron. Identifying non-referential it: A machine learning approach incorporating linguistically motivated patterns. In Proceedings of the ACL Workshop on Feature Engineering for Machine Learning in Natural Language Processing, pages 40-47, Ann Arbor, Michigan, June 2005.

Bresnan, J. 2000. Lexical-Functional Syntax, Blackwell.

Clark A. and Popescu-Belis A. (2004) - Multi-level Dialogue Act Tags. In: Proceedings of SIGDIAL'04 (5th SIGdial Workshop on Discourse and Dialogue) 163-170, Cambridge, MA, USA.

ComLex:- http://nlp.cs.nyu.edu/comlex.

CoreLex:- http://www.cs.brandeis.edu/~paulb/CoreLex/ corelex.html

Delmonte R. 2003. Parsing Spontaneous Speech, in Proc. EUROSPEECH2003, Pallotta Vincenzo, Popescu-Belis Andrei, Rajman Martin "Robust Methods in Processing of Natural Language Dialogues" , Genève, ESCA, 1-6.

Delmonte R., 2005. Parsing Overlaps, in B.Fisseni, H.C.Schmitz, B. Schroeder, P. Wagner (Hrsg.), Sprachtechnologie, mobile Kommunikation und linguistiche Ressourcen, Sprache, Sprechen und Computer, Bd.8, Peter Lang, Frankfurt am Main, 497-512.

Delmonte R., Antonella Bristot, Marco Aldo Piccolino Boniforti, Sara Tonelli, (2005), Modeling Conversational Styles in Italian by means of Overlaps, AISV, CNR, Padova, 11-30.

Delmonte, R. A. Bristot, M. A. Piccolino Boniforti and S. Tonelli, 2006. Another Evaluation of Anaphora Resolution Algorithms and a Comparison with GETARUNS' Knowledge Rich Approach, ROMAND 2006, 11th EACL, Trento, Association for Computational Linguistics, 3-10.

Delmonte R., G. Nicolae, S. Harabagiu. 2007. A Linguistically-based Approach to Detect Causality Relations in Unrestricted Text, in Proc. MICAI-2007, IEEE Publications, pp. 173-185.

Delmonte R. 2007. *Computational Linguistic Text Processing – Logical Form, Semantic Interpretation, Discourse Relations and Question Answering*, Nova Science Publishers, New York.

Delmonte R. 2009. *Computational Linguistic Text Processing – Lexicon, Grammar, Parsing and Anaphora Resolution*, Nova Science Publishers, New York.

Delmonte R., 2008. Semantic and Pragmatic Computing with GETARUNS, in Bos & Delmonte (eds.), STEP, College Pub. London, pp. 287-298.

Richard Evans. 2001. Applying machine learning towardan automatic classification of It. Literary and Linguistic Computing, 16(1):45 – 57.

Fellbaum, Christiane, (ed.) 1998. *WordNet: An Electronic Lexical Database.* MIT Press, Cambridge MA.

Galley, M., McKeown, K., Fosler-Lussier, E. and Jing H. 2003. Discourse Segmentation of Multi-Party Conversation. In Proceedings of ACL 2003, pages 562–569, Sapporo, Japan.

Janin, Adam, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke & Chuck Wooters. 2003. The ICSI Meeting Corpus. Proceedings of IEEE/ICASSP 2003, 6--10 April 2003, Hong Kong, vol. 1, pp. 364--367.

Surabhi Gupta, Matthew Purver and Dan Jurafsky. 2007. Disambiguating Between Generic and Referential "You" in Dialog. Proceedings of ACL 2007 short papers, Prague, Czech Republic.

N. Javanovich and R. op den Akker, Towards Adressee Identification in Multi-party dialogues, in *Proceedings of the 5th Sigdial Workshop on Discourse and Dialogue*, ACL, Pennsylvania, 2004, pp. 89-92.

Goffman E. (1981). Forms of Talk. Philadelphia: University of Pennsylvania Press.

Hillard, D., Ostendorf, M. and Shriberg, E.. 2003. Detection of agreement vs. disagreement in meetings: Training with unlabeled data. In Proceedings of HLT-NAACL 2003.

Kunz W. and Rittel H. W. J. (1970). Issues as elements of information systems. Technical Report WP-131, Berkeley: University of California.

Larsson, S. 2002. Issue-based Dialogue Management. Gothenburg monographs in linguistics 21, University of Goeteborg.

Lo Cascio V. (1991). Grammatica dell'Argomentare: strategie e strutture. Firenze: La Nuova Italia.

Mann, W.C and S.A Thompson. 1988. Rhetorical Structure Theory: Towards a Functional Theory Text Organization. Text, 8(3):243–281.

Christoph Müller. 2007. Resolving *It*, *This*, and *That* in Unrestricted Multi-Party Dialog. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, Prague, Czech Republic, June 23-30, 2007, pp. 816-823.

Christoph Müller. 2006. Automatic Detection on Nonreferential *It* In Spoken Multi-Party Dialog. In: Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, Trento, Italy, April 3-7, 2006, pp. 49-56.

Niekrasz J., Purver M., Dowding J. and Peters S. (2005). Ontology-Based Discourse Understanding for a Persistent Meeting Assistant. In: Proceedings of the AAAI Spring Symposium Persistent Assistants: Living and Working with AI. Stanford.

Pallotta, V., H. Ghorbel, A. Ballim, A. Lisowska and S. Marchand-Maillet. 2004. Towards meeting information systems: Meeting knowledge management. In *Proceedings of ICEIS 2005*, pages 464–469, Porto, Portugal.

Pallotta, V., Seretan V., Ailomaa M. 2007. User requirements analysis for Meeting Information Retrieval based on query elicitation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics* (ACL 2007), pp. 1008–1015, Prague.

Rosemberg, D. and J.A.A. Silince. 1999. Common ground in computer-supported collaborative argumentation. *Paper presented at Workshop on Computer Supported Collaborative Argumentation for Learning Communities* (CSCL99).

Schegloff E. and Sacks H. (1973). Opening up closings. Semiotica 8: 289-327.

Strube, M. and C. Müller, A Machine Learning Approach to Pronoun Resolution in Spoken Dialogue, in *Proceedings of the 41st Annual Meeting of the ACL,* Sapporo, Japan, 7–12 July 2003, pp. 168–175.

APPENDIX I

EPISODES ARE MADE UP OF AGGREGATED UTTERANCES

EPISODE ISSUE No. 1
'Don'-first_1, 'Don'-first_2, 'Don'-first_3

EPISODE ISSUE No. 2
'Morgan'-first_21, 'Morgan'-'Don'-first_22

EPISODE ISSUE No. 3
'Morgan'-first_29, 'Morgan'-first_30, 'Morgan'-first_31, 'Morgan'-first_32

ARGUMENTATIVE STRUCTURE IS BUILT ON THE BASIS OF DIALOGUE ACTS AND DISCOURSE STRUCTURES

```
**********************************************************************
dial_act('Don'-nil, spact, 1, first_1, [ack, bkc, fgb, fhd])[yeah, we, had, a, long, discussion, about, how_much, ..., how, easy, we, want, to, make, it, for, people, to, bleep, things, out, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []),
provide_expl_just(bleep_out, [agent:id18, theme_aff:id14]), provide_expl_just(make, [agent:id18, agent:id19, theme_aff:id20, patient:id13]), provide_expl_just(want, [actor:id18, prop:id21])]
**********************************************************************
dial_act('Don'-nil, spact, 2, first_2, [fgb, fhd])[so, ..., '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, [])]
**********************************************************************
dial_act('Don'-nil, spact, 3, first_3, [])['Morgan', wants, to, make, it, hard, '.']
EVALUATIVE CONTENT:positive, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(make, [agent:id10, theme_aff:id27]), provide_expl_just(want, [actor:id10, prop:id28])]
**********************************************************************
dial_act('Dave'-nil, spact, 4, first_4, [])[it, -, it, 'doesn-t_', ..., '.']
EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[disagree(do, [theme_unaff:id35]), provide_expl_just(be, [])]
**********************************************************************
dial_act('Don'-nil, spact, 5, first_5, [])[did, -, did, -, did, it, ..., ?]
```

EVALUATIVE CONTENT:suspension, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), request_expl_just(be, []), request_expl_just(do, _)]
****

dial_act('Don'-'Dave', after, 6, first_6, [])[i, 'didn-t_', even, check, yesterday, whether, it, was, moving, '.']

EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[disagree(check, [agent:id3, theme_aff:id41, prop:id43]), provide_expl_just(be, [])]
****

dial_act('Dave'-nil, spact, 7, first_7, [])[it, 'didn-t_', move, yesterday, either, when, i, started, it, '.']

EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[disagree(move, [theme_aff:id49])]
****

dial_act('Don'-'Dave', after, 8, first_8, [fgb, fhd])[so, ..., '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, [])]
****

dial_act('Dave'-'Don', after, 9, first_9, [fgb, fhd])[so, i, 'don-t_', know, if, it, 'doesn-t_', like, both, of, us, '.']

EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[disagree(know, [actor:id2])]
****

dial_act('Don'-nil, spact, 10, first_10, [])[channel, three, ?]
EVALUATIVE CONTENT:suspension, ARGUMENTATIVE CONTENT:[request_expl_just(nil, [])]
****

dial_act('Don'-nil, spact, 11, first_11, [])[channel, three, ?]
EVALUATIVE CONTENT:suspension, ARGUMENTATIVE CONTENT:[request_expl_just(nil, [])]
****

dial_act('Dave'-'Adam', after, 12, first_12, [fhd])[you_know, i, discovered, something, yesterday, on, these, um, wireless, ones, '.']
EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(discover, [actor:id70])]
****

dial_act('Adam'-'Don', after, 13, first_13, [ack, bkc])[channel, two, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[accept_expl_opin(nil, [])]
****

dial_act('Don'-nil, spact, 14, first_14, [ack, bkc])[uhhuh, ?]
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), request_expl_just(be, []), request_expl_just(say, [actor:id3, theme_unaff:id86, addressee:id87])]
****

dial_act('Dave'-'Don', after, 15, first_15, [ack, bkc, fgb, fhd])[you_can_tell, if, 'it-s_', picking_up, breath, noise, and, stuff, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(pick, _), provide_expl_just(pick, _), provide_expl_just(pick, _), provide_expl_just(pick, _)]
****

dial_act('Don'-nil, spact, 16, first_16, [ack, bkc, fgb, fhd])[yeah, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[accept_expl_opin(nil, [])]
****

dial_act('Don'-'Dave', while, 17, first_17, [ack, bkc])[it, has, a_little, indicator, on, it, -, on, the, a_f_, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(have, [experiencer:id102, tema_bound:id100])]
****

dial_act('Dave'-nil, spact, 18, first_18, [ack, bkc])[uhhuh, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[accept_expl_opin(nil, [])]
****

dial_act('Dave'-nil, spact, 19, first_19, [fgb, fhd])[so, if, you, ..., '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, [])]
****

dial_act('Dave'-nil, spact, 20, first_20, [ack, bkc, fgb, fhd])[yeah, if, you, breathe, under, -, breathe, and_then, you, see, a_f_, go_off, then, you_know, 'it-s_', ..., picking_up, your, mouth, noise, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(breathe, _), provide_expl_just(go_off, _), provide_expl_just(pick_up, _)]
****

dial_act('Morgan'-nil, spact, 21, first_21, [ack, ass, bkc])[oh, 'that-s_', good, '.']
EVALUATIVE CONTENT:positive, ARGUMENTATIVE CONTENT:[accept_expl_just(be, []), provide_expl_just(be, []), provide_expl_just(be, [prop:infon434])]
****

dial_act('Morgan'-'Don', while, 22, first_22, [bkc, fgb])[because, we, have, a_lot, of, breath, noises, '.']
EVALUATIVE CONTENT:positive, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(have, [prop:infon457])]
****
dial_act('Don'-nil, spact, 23, first_23, [bkc])[yep, '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(say, [actor:id3, theme_unaff:id131, addressee:id132])]
****
dial_act('Don'-nil, spact, 24, first_24, [])[test, '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, [])]
****
dial_act('Morgan'-nil, spact, 25, first_25, [fgb])[in_fact, if, you, listen, to, just, the, channels, of, people, not, talking, 'it-s_', like, ..., '.']
EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[disagree(listen, [agent:id3, theme_aff:id77, result:id140]), provide_expl_just(be, [])]
****
dial_act('Morgan'-'Don', while, 26, first_26, [fgb])['it-s_', very, disgust, ..., '.']
EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[propose_opin_neg(be, [prop:infon535]), provide_expl_just(be, [])]
****
dial_act('Don'-nil, spact, 27, first_27, [fgb])[what, ?]
EVALUATIVE CONTENT:suspension, ARGUMENTATIVE CONTENT:[request_expl_just(nil, [])]
****
dial_act('Don'-'Morgan', while, 28, first_28, [fgb, fhd])[did, you, see, hannibal, recently, or, something, ?]
EVALUATIVE CONTENT:positive, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), request_expl_just(be, []), request_expl_just(see, [experiencer:nil, actor:id153])]
****
dial_act('Morgan'-nil, spact, 29, first_29, [fgb])[sorry, '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, [])]
****
dial_act('Morgan'-nil, spact, 30, first_30, [ack])[exactly, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[accept_expl_opin(nil, [])]
****
dial_act('Morgan'-nil, spact, 31, first_31, [fgb, neg])['it-s_', very, disconcerting, '.']
EVALUATIVE CONTENT:negative, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(be, [prop:infon610])]
****
dial_act('Morgan'-nil, spact, 32, first_32, [ack, bkc, fgb, fhd])[okay, '.']
EVALUATIVE CONTENT:acceptance, ARGUMENTATIVE CONTENT:[accept_expl_opin(nil, [])]
****
dial_act('Morgan'-'Don', after, 33, first_33, [fgb, fhd])[so, um, ..., '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, [])]
****
dial_act('Don'-nil, spact, 34, first_34, [disf])[h, ..., '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, [])]
****
dial_act('Morgan'-nil, spact, 35, first_35, [])[i, was, going, to, try, to, get_out_of, here, like, in, half, an, hour, '.']
EVALUATIVE CONTENT:suspension, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), suggest(get_out_of, [agent:id10, theme_aff:id180]), suggest(try, [actor:id10, prop:id181])]
****
dial_act('Morgan'-nil, spact, 36, first_36, [fgb, fhd])[um, ..., '.']
EVALUATIVE CONTENT:none, ARGUMENTATIVE CONTENT:[provide_expl_just(be, []), provide_expl_just(say, [actor:id10, theme_unaff:id187, addressee:id188])]
****
dial_act('Morgan'-nil, spact, 37, first_37, [fgb])[because, i, really, appreciate, people, coming, '.']
EVALUATIVE CONTENT:positive, ARGUMENTATIVE CONTENT:[provide_expl_just(appreciate, [agent:id10, theme_aff:id13]), provide_expl_just(be, [])]