Chapter 1

MIND: A CONTEXT-BASED MULTIMODAL INTERPRETATION FRAMEWORK IN CONVERSATIONAL SYSTEMS

Joyce Y. Chai
Department of Computer Science and Engineering
Michigan State University, East Lansing, Michigan 48824, USA
jchai@cse.msu.edu

Shimei Pan and Michelle X. Zhou
IBM T. J. Watson Research Center
19 Skyline Drive, Hawthorne, NY 10532, USA
{shimei, mzhou}@us.ibm.com

Abstract In a multimodal human-machine conversation, user inputs are often abbreviated or imprecise. Simply fusing multimodal inputs together may not be sufficient to derive a complete understanding of the inputs. Aiming to handle a wide variety of multimodal inputs, we are building a context-based multimodal interpretation framework called MIND (Multimodal Interpreter for Natural Dialog). MIND is unique in its use of a variety of contexts, such as domain context and conversation context, to enhance multimodal interpretation. In this chapter, we first describe a fine-grained semantic representation that captures salient information from user inputs and the overall conversation, and then present a context-based interpretation approach that enables MIND to reach a full understanding of user inputs, including those abbreviated or imprecise ones.

Keywords: Multimodal input interpretation, multimodal interaction, conversation systems.

1. Introduction

To aid users in their information-seeking process, we are building an infrastructure, called Responsive Information Architect (RIA), which can engage users in an intelligent multimodal conversation. Currently, RIA is embodied in

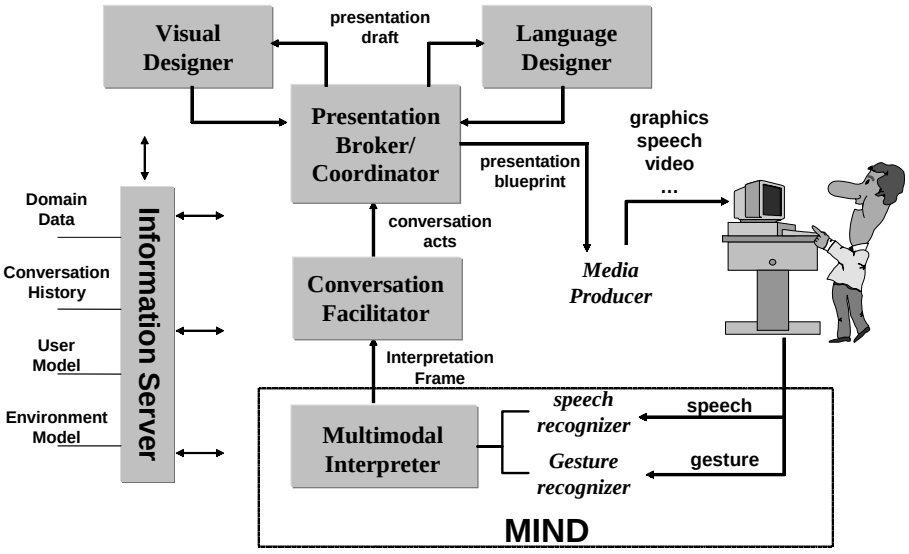

Figure 1.1. RIA infrastructure.

a testbed, called Real Hunter^{TM}, a real-estate application for helping users find residential properties.

Figure 1.1 shows RIA's main components. A user can interact with RIA using multiple input channels, such as speech and gesture. To understand a user input, the multimodal interpreter exploits various contexts (e.g., conversation history) to produce an interpretation frame that captures the meanings of the input. Based on the interpretation frame, the conversation facilitator decides how RIA should act by generating a set of conversation acts (e.g., `Describe` information to the user). Upon receiving the conversation acts, the presentation broker sketches a presentation draft that expresses the outline of a multimedia presentation. Based on this draft, the language and visual designers work together to author a multimedia blueprint which contains the details of a fully coordinated multimedia presentation. The blueprint is then sent to the media producer to be realized. To support all components described above, an information server supplies various contextual information, including domain data (e.g., houses and cities for a real-estate application), a conversation history (e.g., detailed conversation exchanges between RIA and a user), a user model (e.g., user profiles), and an environment model (e.g., device capabilities).

Our focus in this chapter is on the interpretation of multimodal user inputs. We are currently developing a context-based multimodal interpretation framework called MIND (Multimodal Interpreter for Natural Dialog). MIND is inspired by the earlier works on input interpretation from both multimodal

interaction systems, e.g., [Bolt, 1980; Burger and Marshall, 1993; Cohen et al., 1997; Zancanaro et al., 1997; Wahlster, 1998; Johnston and Bangalore, 2000] and spoken dialog systems [Allen et al., 2001; Wahlster, 2000]. Specifically, MIND presents two unique features. First, MIND exploits a fine-grained semantic model that characterizes the meanings of user inputs and the overall conversation. Second, MIND employs an integrated interpretation approach that uses a wide variety of contexts (e.g., conversation history and domain knowledge). These two features enable MIND to enhance understanding of user inputs, including those ambiguous and incomplete inputs.

2. Related Work

Since the first appearance of the "Put-That-There" system [Bolt, 1980], a variety of multimodal systems have emerged, from earlier versions that combined speech, mouse pointing [Neal and Shapiro, 1988], and gaze [Koons et al., 1993], to systems that integrate speech with pen based gestures, e.g., hand drawn graphics [Cohen et al., 1997; Wahlster, 1998]. There are also more sophisticated systems that combine multimodal inputs and outputs [Cassell et al., 1999], and those that work in a mobile environment [Johnston et al., 2002; Oviatt, 2000]. Recently, we have seen a new generation of systems that not only support multimodal user inputs, but can also engage users in an intelligent conversation [Alexandersson and Becker, 2001; Gustafson et al., 2000; Johnston et al., 2002]. To function effectively, each of these systems must be able to adequately interpret multimodal user inputs. Substantial work on multimodal interpretation has been focusing on semantic fusion [Johnston, 1998; Johnston and Bangalore, 2000; Vo and Wood, 1996; Wu et al., 1999]. In contrast, this chapter describes a framework that combines semantic fusion with context-based inference for multimodal interpretation.

3. MIND Overview

To interpret multimodal user inputs, MIND takes three major steps as shown in Figure 1.2: unimodal understanding, multimodal understanding, and discourse understanding. During unimodal understanding, MIND applies modality specific recognition and understanding components (e.g., a speech recognizer and a language interpreter) to identify meanings of each unimodal input. During multimodal understanding, MIND combines semantic meanings of unimodal inputs and uses contexts (e.g., conversation context and domain context) to form an overall understanding of multimodal user inputs. Furthermore, MIND also identifies how an input relates to the overall conversation discourse through discourse understanding.

In particular, MIND groups together inputs that contribute to the same goal/sub-goal [Grosz and Sidner, 1986]. The result of discourse understand-

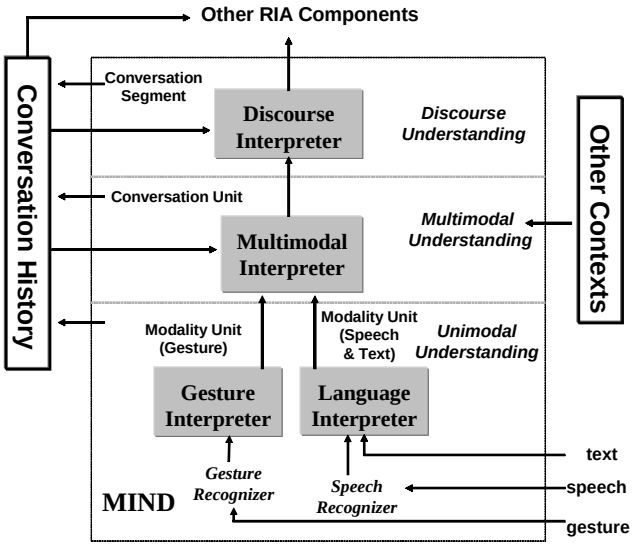

Figure 1.2. MIND overview.

ing is an evolving conversation history that reflects the overall progress of a conversation.

4. Example Scenario

Figure 1.3 shows a conversation fragment between a user and RIA. The user initiates the conversation by asking for houses in Irvington (U1), and RIA replies by showing a group of desired houses (R1). Based on the generated visual display, the user points to the screen (a position between two houses as in Figure 1.4) and asks for the price (U2). In this case, it is not clear which object the user is pointing at. There are three candidates: two houses nearby and the town of Irvington. Using our domain knowledge, MIND can rule out the town of Irvington, since he is asking for a price. At this point, MIND still cannot determine which of the two house candidates is intended. To clarify this ambiguity, RIA highlights both houses and asks the user to pinpoint the house of his/her interest (R2).

Again, the user's reply (U3) alone would be ambiguous, since there are multiple green objects on the screen. However, using the conversation history and the visual properties, MIND is able to infer that s/he is referring to the highlighted green house. The user continues on to ask for the size (U4). This request by itself is incomplete, since s/he did not explicitly specify the object of his/her interest (house). Nevertheless, MIND understands that the user is asking for the size of the same green house based on the conversation context.

User (U1):	Speech: Show me houses here Gesture: point to Irvington on the screen
RIA (R1):	Speech: Here are the houses you requested Graphics: show a set of house icons on the screen
User (U2):	Speech: How much is this? Gesture: point to the screen (not directly on any object)
RIA (R2):	Speech: Which house are you interested in? Graphics: highlight both house candidates
User (U3):	Speech: The green one.
RIA (R3):	Speech: The green house costs 250,000 dollars Graphics: highlight the green house
User (U4):	Speech: How large?
RIA (R4):	Speech: The size of this house is 2500 square feet.
User (U5):	Speech: What about this one? Gesture: point to another house
RIA (R5):	Speech: The size of this house is 2200 square feet Graphics: highlight another house
User (U6):	Speech: Show me houses with this style around here. Gesture: point to east of Irvington on the map
RIA (R6):	Speech: I found seven Victorian style houses in White Plains Graphics: show seven houses icons in White Plains

Figure 1.3. Example scenario.

The user moves on to inquire about another house (U5). This input by itself does not indicate exactly what the user wants. Again, using the conversation context, MIND recognizes that s/he is most likely asking for the size of another house. Finally, in U6, the user asks for houses with a certain style in a certain location. Although there is only one deictic gesture, based on the visual context, MIND understands that "this style" refers to the style of the highlighted house on the screen and "here" refers to the location of White Plains, which is east of Irvington.

As mentioned above, our example shows that multimodal user inputs exhibit a wide range of varieties. The inputs may be abbreviated, ambiguous or complex, and simply fusing inputs together often cannot reach a full understanding. For example the deictic gesture in U2 is ambiguous. It is not clear which object the user is pointing at: two houses nearby or the town of Irvington. The user input U5 by itself is incomplete, since the purpose of the input is unspecified. Furthermore, in U6, a single deictic gesture overlaps (in terms of time) with both "this style" and "here" from the speech input; it is hard to determine which one of those two references should be aligned and fused with the gesture. Processing these inputs requires context to be taken into consideration. Therefore, we have designed and implemented a context-based interpretation approach in MIND. Currently, MIND uses three types of contexts: domain context, conversation context, and visual context. The do-

Figure 1.4. Imprecise pointing on the graphic display.

main context provides domain knowledge. The conversation context reflects the progress of the overall conversation. The visual context gives the detailed semantic and syntactic structures of visual objects and their relations. In the next few sections, we first describe our semantics-based representation, and then present the context-based approach using this representation.

5. Semantics-Based Representation

To support context-based multimodal interpretation, we need to represent both user inputs and various contextual information. In this chapter, we focus on describing the representations of user inputs and the conversation context. In particular, we discuss two aspects of the representation: semantic models that capture salient information and structures that represent these semantic models.

5.1 Semantic Modelling of User Inputs

To model user inputs, MIND has two goals. First, MIND must understand the meanings of user inputs so that the conversation facilitator (Figure 1.1) can decide how the system should act. Second, MIND should capture the user input styles (e.g., using a particular verbal expression or gesturing in a particular way) or user communicative preferences (e.g., preferring a verbal vs. a visual presentation) so that such information can help the multimedia generation components (visual or language designers in Figure 1.1) to create more effective and tailored system responses. To accomplish both goals, MIND characterizes four aspects of a user input: intention, attention, presentation preference, and interpretation status.

5.1.1 Intention. `Intention` describes the purpose of a user input [Grosz and Sidner, 1986]. We further characterize three aspects of an intention: `Motivator`, `Act`, and `Method`. `Motivator` captures the purpose of an interaction. Since we focus on information-seeking applications, MIND currently distinguishes three top-level purposes: `DataPresentation`, `DataAnalysis` (e.g., comparison), and `ExceptionHandling` (e.g., disambiguation). `Act` indicates one of the three user actions: request, reply, and inform. `Request` specifies that a user is making an information request (e.g., asking for a collection of houses in U1). `Reply` indicates that the user is responding to a previous RIA request (e.g., confirming the house of interest in U3). Unlike `Request` or `Reply`, `Inform` states that a user is simply providing RIA with specific information, such as personal profiles or interests. For example, during a house exploration, a user may tell RIA that she has school-age children.

`Method` further refines a user action. For example, MIND may distinguish two different types of requests. One user may request RIA to `Describe` the desired information, such as the price of a house, while the other may request RIA simply to `Identify` the desired information (e.g., show a train station on the screen).

In some cases, `Motivator`, `Act` and `Method` can be directly captured from individual inputs (e.g. U1). However, in other situations, they can only be inferred from the conversation discourse. For example, from U3 itself, MIND only understands that the user is referring to a green object. It is not clear whether this is a reply or an inform. Moreover, the overall purpose of this input is also unknown. Nevertheless, based on the conversation context, MIND understands that this input is a reply to a previous RIA question (`Act`: `Reply`), and contributes to the overall purpose of an exception handling intent (`Motivator`: `ExceptionHandling`).

In addition to recording the purpose of each user input, `Motivator` also captures discourse purposes (described later). Therefore, `Motivator` can be also viewed as characterizing sub-dialogues discussed in previous literatures [Lambert and Carberry, 1992; Litman and Allen, 1987]. For example, `ExceptionHandling` (with `Method`: Correct) corresponds to a Correction sub-dialogue. However, unlike earlier works, our `Motivator` is used to model intentions at both input (turn) and discourse levels. Finally, we model intention not only to support the understanding of a conversation, but also to facilitate the multimedia generation. Specifically, `Motivator` and `Method` together direct RIA in its response generation. For example, RIA would consider `Describe` and `Identify` two different data presentation directives [Zhou and Pan, 2001].

Figure 1.5(a) shows the `Intention` that MIND has identified from the user input U2 (Figure 1.3). It says that the user is asking RIA to present him with desired information, which is captured in `Attention` below.

(a) Intention	(b) Attention	(c) Presentation Preference	(d) Interpretation Status
Motivator: DataPresentation **Act**: Request **Method**: Describe	**Base**: House **Topic**: Instance **Focus**: SpecificAspect **Aspect**: Price **Content**: <MLS0187652 | MLS0889234>	**Directive**: Summary **Media**: Multimedia **Device**: Desktop **Style**: < >	**SyntacitcCompleteness**: ContentAmbiguity **SemanticCompleteness**: Yes

Figure 1.5. Semantic modelling for user inputs.

5.1.2 Attention. While `Intention` indicates the purpose of a user input, `Attention` captures the content of a user input with six dimensions. `Base` specifies the semantic category of the content (e.g., all houses in our application belong to the `House` category). `Topic` indicates whether the user is concerned with a concept, a relation, an instance, or a collection of instances. For example, in U1 (Figure 1.3) the user is interested in a collection of `House`, while in U2 he is interested in a specific instance. `Focus` further narrows down the scope of the content to distinguish whether the user is interested in a topic as a whole or just the specific aspects of the topic. For example, in U2 the user focuses only on one specific aspect (*price*) of a house instance. `Aspect` enumerates the actual topical features that the user is interested in (e.g., the price in U2). `Constraint` holds the user constraints or preferences placed on the topic. For example, in U1 the user is only interested in the house instances (`Topic`) located in Irvington (`Constraint`). The last parameter `Content` points to the actual data in our database.

Figure 1.5(b) records the `Attention` identified by MIND for the user input U2. It states that the user is interested in the price of a house instance, MLS0187652 or MLS0889234 (house ids from the Multiple Listing Service). As discussed later, our fine-grained modelling of `Attention` provides MIND the ability to discern subtle changes in user interaction (e.g., a user may focus on one topic but explore different aspects of the topic). This in turn helps MIND assess the overall progress of a conversation.

5.1.3 Presentation preference. During a human-computer interaction, a user may indicate what type of responses she prefers. Currently, MIND captures user preferences along four dimensions. `Directive` specifies the high-level presentation goal (e.g., preferring a summary to details). `Media` indicates the preferred presentation medium (e.g., verbal vs. visual). `Style` describes what general formats should be used (e.g., using a chart vs. a diagram to illustrate information). `Device` states what devices would be used in the presentation (e.g., phone or PDA). Using the captured presentation preferences, RIA can generate multimedia presentations that are tailored to individual users and their goals. For example, Figure 1.5(c) records the user preferences from U2.

Since the user did not explicitly specify any preferences, MIND uses the default values to represent those preferences. Presentation preferences can either be directly derived from user inputs or inferred based on user and environment contexts.

5.1.4 Interpretation status. Interpretation status provides an overall assessment on how well MIND understands an input. This information is particularly helpful in guiding RIA's next move. Currently, it includes two features. `SyntacticCompleteness` assesses whether there is any unknown or ambiguous information in the interpretation result. `SemanticCompleteness` indicates whether the interpretation result makes sense. Using the status, MIND can inform other RIA components whether a certain exception has risen. For example, the value `ContentAmbiguity` in `SyntacticCompleteness` (Figure 1.5d) indicates that there is an ambiguity concerning `Content` in `Attention`, since MIND cannot determine whether the user is interested in MLS0187652 or MLS0889234. Based on this status, RIA would ask a clarification question to disambiguate the two houses (e.g., R2 in Figure 1.3).

5.2 Semantic Modelling of Conversation Discourse

In addition to modelling the meanings of user inputs at each conversation turn, we also model the overall progress of a conversation. Based on Grosz and Sidner's conversation theory [Grosz and Sidner, 1986], MIND establishes a refined discourse structure as conversation proceeds. This is different from other multimodal systems that maintain the conversation history by using a global focus space [Neal et al., 1998], segmenting a focus space based on intention [Burger and Marshall, 1993], or establishing a single dialogue stack to keep track of open discourse segments [Stent et al., 1999].

5.2.1 Conversation unit and segment. Our discourse structure has two main elements: conversation units and conversation segments. A conversation unit records user or RIA actions at a single turn of a conversation. These units can be grouped together to form a segment (e.g., based on their intentional similarities). Moreover, different segments can be organized into a hierarchy (e.g., based on intentions and sub-intentions). Figure 1.6 depicts the discourse structure of the conversation after interpreting U3 in Figure 1.2. This structure contains five conversation units (rectangles U1–3 for the user, R1–2 for RIA) and three conversation segments (ovals DS1–3).

A user conversation unit contains the interpretation result of a user input discussed in the last section (as shown in U1). A RIA unit contains the automatically generated multimedia response, including the semantic and syntactic structures of a multimedia presentation [Zhou and Pan, 2001]. A conversation

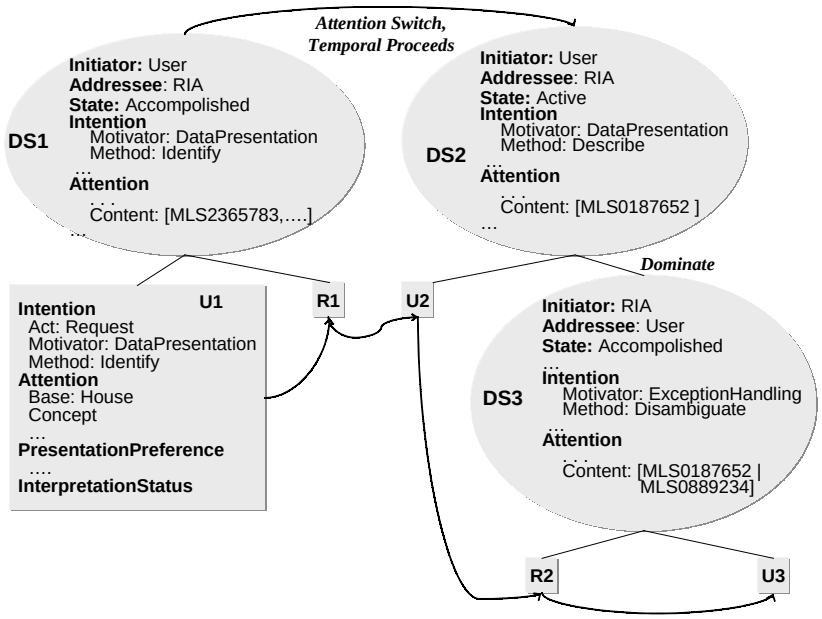

Figure 1.6. Conversation discourse.

segment has five features: `Initiator`, `Addressee`, `State`, `Intention`, and `Attention`. The `Intention` and `Attention` are similar to those modelled in the units (see DS1 and U1). `Initiator` indicates the conversation initiating participant (e.g., `Initiator` is User in DS1), while `Addressee` indicates the recipient of the conversation (e.g., `Addressee` is RIA in DS1). Currently, we are focused on one-to-one conversations. However, MIND can be extended to support multi-party conversations where `Addressee` could be a group of agents. Finally, `State` reflects the current state of a segment: active, accomplished or suspended. For example, after interpreting U3, DS2 is still active (before R2 is generated), but DS3 is already accomplished, since its purpose of disambiguating the content has been fulfilled.

5.2.2 Discourse relations. To model the progress in a conversation, MIND captures three types of discourse relations: conversation structural relations, conversation transitional relations, and data transitional relations.

Conversation structural relations reveal the intentional structure between the purposes of conversation segments. Following Grosz and Sidner's early work, there are currently two types: dominance and satisfaction-precedence. For example, in Figure 1.6, DS2 dominates DS3, since the segment of `ExceptionHandling` (DS3) is for the purpose of `DataPresentation` (DS2).

Conversation transitional relations specify transitions between conversation segments and between conversation units as the conversation unfolds. Currently, two types of relations are identified between segments: intention switch and attention switch. The intention switch relates two segments, which differ in their intentions. Interruption is a sub-type of an intention switch. The attention switch relates two segments, which possess the same intention but differ in their attention. For instance, in Figure 1.6, there is an attention switch from DS1 to DS2 since DS1 concerns about a collection of houses and DS2 focuses on one particular house. Furthermore, there are also temporal-precedence relations that connect different segments together based on the order when they occur. The temporal-precedence relation also connects conversation units to preserves the sequence of conversation.

Data transitional relations further discern different types of attention switches. In particular, we distinguish eight types of attention switches, including `Collection-to-Instance` and `Instance-to-Aspect`. For example, the attention is switched from a collection of houses in DS1 to a specific house in DS2 (Figure 1.6). Data transitional relations allow MIND to capture user data exploration patterns. Such patterns in turn can help RIA decide potential data navigation paths and provide users with an efficient information-seeking environment.

Our studies showed that, in an information-seeking environment, the conversation flow usually centres around the data transitional relations. This is different from task-oriented applications where dominance and satisfaction precedence are greatly observed. In an information seeking application, the communication is more focused on the type and the actual content of information, which often by itself does not impose any dominance or precedence relations.

5.3 Representing Intention and Attention in Feature-Based Structures

Based on the semantic models of intention and attention described earlier, MIND uses feature-based structures to represent intention and attention. The type of a structure is captured by a special feature `FsType`. In an `Intention` structure, `FsType` takes the value of `Motivator` (i.e., the purpose of communication), and in an `Attention` structure, `FsType` takes the value of `Base` (i.e., the semantic category of the content). Since the values of `Motivator` or `Base` may not be able to be inferred from current user inputs directly, we have added a value `Unknown`. In addition to other characteristics of using feature-based structures [Carpenter, 1992], our representation has two unique features.

The first characteristic is that intention and attention are consistently represented at different processing stages. More specifically, MIND uses the same

set of features to represent intention and attention identified from unimodal user inputs, combined multimodal inputs, and conversation segments. Figure 1.7(a) outlines the intention and attention identified by MIND for the speech input in U2. Since the semantic type of the content is unknown, the type of the `Attention` is set to unknown (`FsType`: `Unknown`). Note that here we only include the features that can be instantiated. For example feature `Content` is not present in the `Attention` structure, since the exact object of the instance is not specified in the speech input.

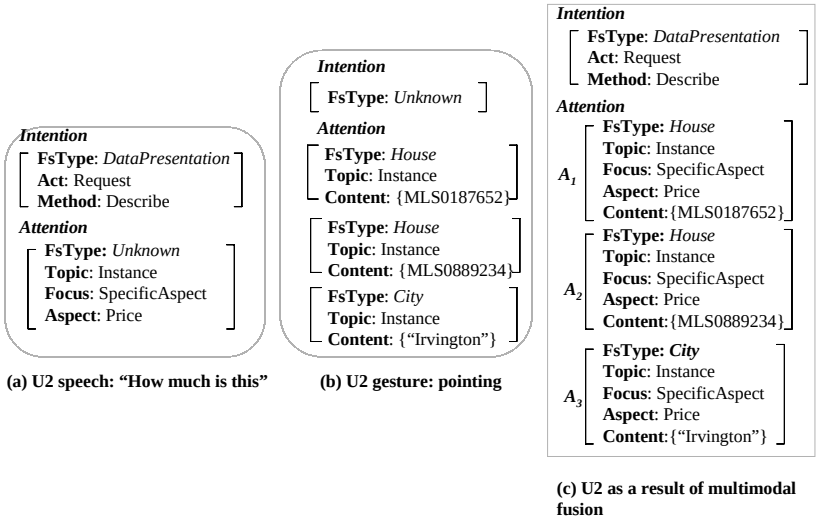

Figure 1.7. Feature structures for intention and attention.

Similarly, we represent the semantic information extracted from a deictic gesture. Figure 1.7(b) shows the corresponding feature structures for U2 gesture input. The `Intention` structure has an `Unknown` type since the high level purpose and the specific task cannot be identified from the gesture input. Furthermore, because of the ambiguity of the deictic gesture, three `Attention` structures are identified. The first two are for house instances MLS0187652 and MLS0889234, and the third is for the town of Irvington. MIND performs multimodal fusion by combining each modality's feature structures into a single feature structure. The result of multimodal fusion is shown in Figure 1.7(c).

Furthermore, MIND uses the same kind of feature structures to represent intention and attention in conversation segments. Figure 1.8 records the conversation segments that cover U2 through R3. As described later, such a consistent representation facilitates a context-driven inference.

The second characteristic of our representation is its flexible composition. One feature structure can be nested in another feature structure. For example, U6 in Figure 1.2 is a complex input, where the speech input "show me houses

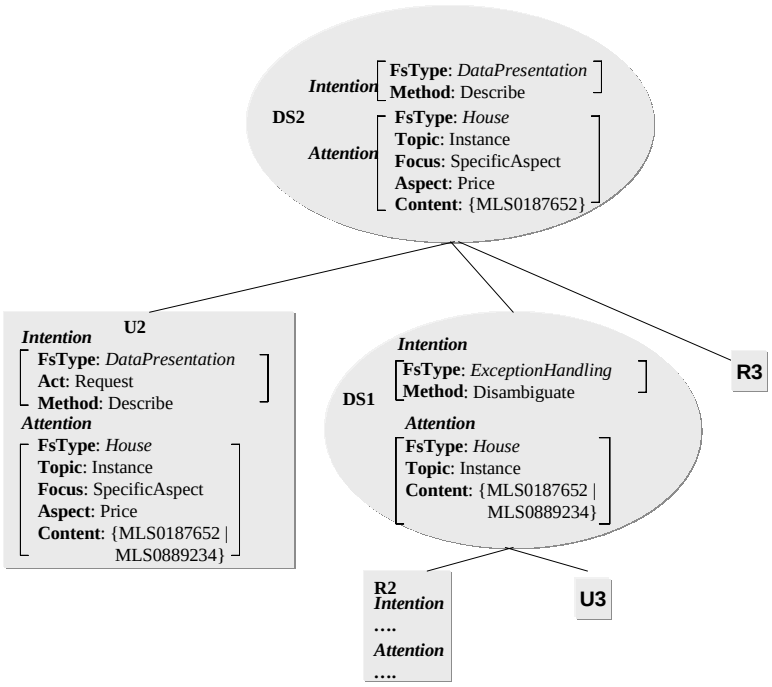

Figure 1.8. Feature structures for intention and attention in conversation segments.

with this style around here" consists of two references. The `Attention` structure created for U6 speech input is shown in Figure 1.9. The structure A_1 indicates that the user is interested in a collection of houses that satisfy two constraints. The first constraint is about the style (`Aspect: Style`), and the second is about the location. Both of these constraints are related to other objects, which are represented by similar `Attention` structures A_2 and A_3 respectively. During the interpretation process, MIND first tries to resolve these two references and then reformulates the overall constraints [Chai, 2002b].

6. Context-Based Multimodal Interpretation

Based on the semantic representations described above, MIND uses a wide variety of contexts to interpret the rich semantics of user inputs. Currently, we support three input modalities: speech, text, and gesture. Specifically, we use IBM ViaVoice to perform speech recognition, and a statistics-based natural language understanding component [Jelinek et al., 1994] to process natural language sentences. For gestures, we have developed a simple geometry-based gesture recognition and understanding component. Based on the output from

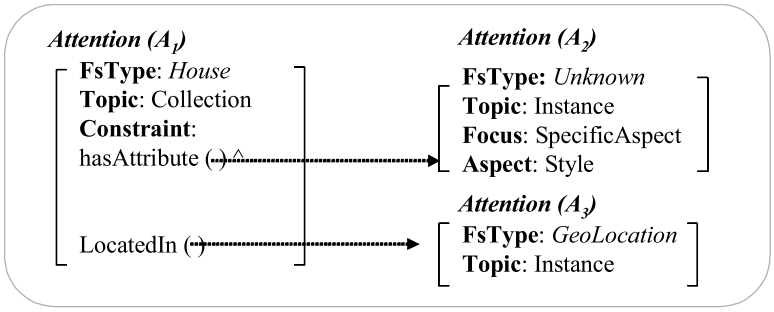

Figure 1.9. `Attention` structures for U6 speech input.

unimodal understanding, MIND performs multimodal understanding, consisting of two sub-processes: multimodal fusion and context based inference.

6.1 Multimodal Fusion

During multimodal fusion, MIND first uses temporal constraints to align `Intention`/`Attention` structures identified from each modality, and then to unify the corresponding structures.

The formally defined unification operation provides a mechanism to combine information from a number of distinct sources into a unique representation [Carpenter, 1992]. Two feature structures can be unified if they have compatible types based on an inheritance hierarchy (an `Unknown` type always subsumes other known types), and the values of the same features are also consistent with each other (i.e., satisfying an inheritance hierarchy). Otherwise, unification fails. Based on this nature, unification is applied in multimodal fusion since information from different modalities tends to complement each other [Johnston, 1998]. MIND also applies unification to multimodal fusion. For example, in Figure 1.7(b), there are three `Attention` structures for U2 gesture input due to the imprecise pointing. Each of them can be unified with the `Attention` structure from U2 speech input (in Figure 1.7a). The fusion result is shown in Figure 1.7(c).

In this case, an ambiguity arises due to the post-fusion presence of three `Attention` structures. In many other cases, the overall meanings of a user input still cannot be identified as a result of multimodal fusion. For example, the exact focus of attention for U4 is still unclear after the fusion. To enhance interpretation, MIND uses various contexts to make inferences about the inputs.

Covering feature structure A onto feature structure B creates feature structure C in following steps:
(1) **if** (the type of A and the type of B are not compatible)
  then covering fails
  else if the type of A is `Unknown`
    assign the type of B to the type of C
  else
    assign the type of A to the type of C
(2) **for** (each feature f_i in both A and B) {
    Suppose u_i is the value in A, v_i is the value in B, and w_i is the value in C
  (a) **if** u_i and v_i are feature structures
    then covering u_i onto v_i and assign the result to w_i // *recursively applying covering*
  (b) **else** assign u_i to w_i // *the value in the covering structure prevails*
(3) **for** (each feature in A but not in B)
  add this feature and its value in C
(4) **for** (each feature in B but not in A)
  add this feature and its value in C

Figure 1.10. Covering operation.

6.2 Context-Based Inference

Currently, MIND uses three types of context: conversation context, domain context, and visual context.

6.2.1 Conversation context. Conversation context provides an overall history of a conversation as described earlier. In an information-seeking environment, users tend to only explicitly or implicitly specify the new or changed aspects of the information of their interest without repeating what has been mentioned earlier in the conversation. Given a partial user input, required but unspecified information needs to be inferred from the conversation context. Currently, MIND applies an operation, called *covering*, to draw inferences from the conversation context [Chai, 2002a]. Although the mechanism of our covering operation is similar to the overlay operation described in [Alexandersson and Becker, 2001], not only can our covering infer the focus of attention (as overlay does), but it can also infer the intention. What makes this operation possible is our underlying consistent representation of intention and attention at both the discourse and the input levels.

Covering combines two feature structures by placing one structure (covering structure) over the other one (covered structure). Figure 1.10 describes the steps of the covering operation. Specifically, if the types of two structures are not compatible, then the covering fails (step 1). For the same features in both structures, the values from the covering structure always prevail and are included in the resulting structure (step 2b). Covering is recursive (step

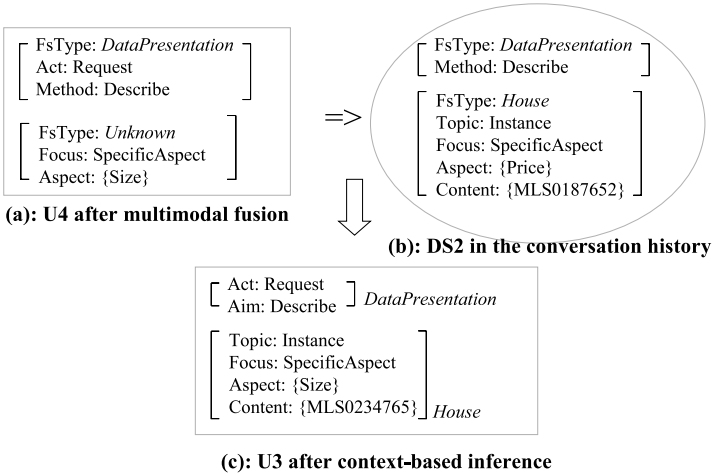

Figure 1.11. Example of context-based inference using covering.

2a). Features that exist only in one structure but not in the other are included automatically in the resulting structure (steps 3 and 4).

For example, to interpret U4, MIND applies the operation by covering U4 on DS2. As a result, features in DS2 (`Topic` and `Content`) are added in the combined structure and the value of the `Aspect` feature is changed to Price (Figure 1.11). Thus, MIND is able to figure out that the user is interested in the size of the same house as in U2. Note that it is important to maintain a hierarchical conversation history based on goals and sub-goals. Without such a hierarchical structure, MIND would not be able to infer the content of U4. Furthermore, because of the consistent representation of intention and attention at both the discourse level (in conversation segments) and the input level (in conversation units), MIND is able to directly use the conversation context to infer unspecified information. The covering operation can also be applied to U5 and the discourse segment built after R4 is processed. Therefore, MIND can infer that the user is asking for the size of another house in U5.

6.2.2 Domain context. Domain context provides the domain knowledge such as semantic and meta information about the application data. The domain context is particularly useful in resolving input ambiguities. For example, to resolve the ambiguity of whether the attention is a city object or a house object (as in U2), MIND uses the domain context. In this case, A_3 in Figure 1.7(c) indicates the interest is the price of the city Irvington. Based on the domain knowledge that the city object does not have a price attribute, A_3 is filtered out. As a result, both A_1 and A_2 are potential interpretations, and

RIA is able to arrange the follow-up question to further disambiguate the two houses (R2 in Figure 1.3).

6.2.3 Visual context. As RIA provides a rich visual environment for users to interact with, users may refer to objects on the screen by their spatial (e.g., the house at the left corner) or perceptual attributes (e.g., the red house). To resolve these spatial/perceptual references, MIND exploits the visual context, which logs the detailed semantic and syntactic structures of visual objects and their relations. More specifically, the automated generated visual encoding for each object is maintained as a part of the system conversation unit in our conversation history. During reference resolution, MIND would identify potential candidates by mapping the referring expressions with the internal visual representation.

For example, the object which is highlighted on the screen (R5) has an internal representation that associates the visual property Highlight with an object identifier. This allows MIND to correctly resolve referents for "this style" in U6. The representation for U6 speech input in Figure 1.9 indicates three attention structures. The gesture input overlaps with both "this style" (corresponding to A_2) and "here" (corresponding to A_3); there is no obvious temporal relation indicating which of these two references should be unified with the deictic gesture. In fact, both A_2 and A_3 are potential candidates. An earlier study [Kehler, 2000] indicates that objects in the visual focus are often referred by pronouns or demonstratives, rather than by full noun phrases or deictic gestures. Based on this finding and using the visual context, MIND infers that most likely "this style" refers to the style of the highlighted house and the deictic gesture resolves the referent of "here". Suppose that the style is Victorian and "here" refers to White Plains, MIND is then able to reformulate the constraints and figure out that the overall meaning of U6: looking for houses with a Victorian style and located in White Plains.

During the context-based inference, MIND applies an instance-based approach to first determine whether there is enough information collected from the current user inputs. We have collected a set of instances, each of which is a pair of valid intention and attention structures that MIND can handle. By comparing the fused representation (the result of the multimodal fusion process) with the set of instances, MIND determines whether the information is adequate for further reasoning. If the information is sufficient, MIND then uses the domain context and visual context to further resolve ambiguities and validate the fused representation. If the information is inadequate, MIND applies the covering operation on conversation segments (starting from the most recent one) to infer the unspecified information from the conversation context.

7. Discussion

In a conversation setting, user inputs could be ambiguous, abbreviated, or complex. Simply fusing multimodal inputs together may not be sufficient to derive a complete understanding of the inputs. Therefore, we have designed and implemented a context-based framework, MIND, for multimodal interpretation. MIND relies on a fine-grained semantic representation that captures salient information from user inputs and the overall conversation. The consistent representation of intention and attention provides a basis for MIND to unify the multimodal fusion and context-based inference processes.

Our ongoing work is focused on the following aspects. First of all, we continue to improve multimodal interpretation by incorporating a probabilistic model [Chai et al., 2004]. Our existing approach of fusion and inference is straightforward for simple user inputs. However, it may be complicated when multiple attentions from one input need to be unified with multiple attentions from another input. Suppose that the user says "tell me more about the red house, this house, the blue house," and at the same time she points to two positions on the screen sequentially. This alignment can be easily performed when there is a clear temporal binding between a gesture and a particular phrase in the speech. However, in a situation where a gesture is followed (preceded) by a phrase without an obvious temporal association as in "tell me more about the red house (deictic gesture 1) this house (deictic gesture 2) the blue house," a simple unification cannot solve the problem.

Furthermore, we are investigating the use of other contexts, such as user context. The user context provides MIND with user profiles. A user profile is established through two means: explicit specification and automated learning. Using a registration process, information about user preferences can be gathered such as whether the school district is important. In addition, MIND can also learn user vocabularies and preferences based on real sessions. One attempt is to use this context to map fuzzy terms in an input to precise query constraints. For example, the interpretation of the terms, such as "expensive" or "big", may vary greatly from one user to another. Based on different user profiles, MIND can interpret these fuzzy terms as different query constraints.

The third aspect is improving discourse interpretation. Discourse interpretation identifies the contribution of user inputs toward the overall goal of a conversation. During the discourse interpretation, MIND decides whether the input at the current turn contributes to an existing segment or starts a new one. In the latter case, MIND must decide where to add the new segment and how this segment relates to existing segments in the conversation history.

Acknowledgements

We would like to thank Keith Houck for his contributions on training models for speech/gesture recognition and natural language parsing, and Rosario Uceda-Sosa for her work on RIA information server.

References

Alexandersson, J. and Becker, T. (2001). Overlay as the Basic Operation for Discourse Processing in a Multimodal Dialogue System. In *Proceedings of the IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Seattle, Washington, USA.

Allen, J., Byron, D., Dzikovska, M., Ferguson, G., Galescu, L., and Stent, A. (2001). Towards Conversational Human-Computer Interaction. *AI Magazine*, 22(4):27–37.

Bolt, R. A. (1980). Voice and Gesture at the Graphics Interface. *Computer Graphics*, pages 262–270.

Burger, J. and Marshall, R. (1993). The Application of Natural Language Models to Intelligent Multimedia. In Maybury, M., editor, *Intelligent Multimedia Interfaces*, pages 429–440. Menlo Park, CA: MIT Press.

Carpenter, B. (1992). *The Logic of Typed Feature Structures*. Cambridge University Press.

Cassell, J., Bickmore, T., Billinghurst, M., Campbell, L., Chang, K., Vilhjálmsson, H., and Yan, H. (1999). Embodiment in Conversational Interfaces: Rea. In *Proceedings of Conference on Human Factors in Computing Systems (CHI)*, pages 520–527, Pittsburgh, PA.

Chai, J. (2002a). Operations for Context-Based Multimodal Interpretation in Conversational Systems. In *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, pages 2249–2252, Denver, Colorado, USA.

Chai, J. (2002b). Semantics-Based Representation for Multimodal Interpretation in Conversational Systems. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING)*, pages 141–147, Taipei, Taiwan.

Chai, J., Hong, P., and Zhou, M. (2004). A Probabilistic Approach for Reference Resolution in Multimodal User Interfaces. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI)*, pages 70–77, Madeira, Portugal. ACM.

Cohen, P., Johnston, M., McGee, D., Oviatt, S., Pittman, J., Smith, I., Chen, L., and Clow, J. (1997). Quickset: Multimodal Interaction for Distributed Applications. In *Proceedings of the Fifth Annual International ACM Multimedia Conference*, pages 31–40, Seattle, USA.

Grosz, B. J. and Sidner, S. (1986). Attention, Intentions, and the Structure of Discourse. *Computational Linguistics*, 12(3):175–204.

Gustafson, J., Bell, L., Beskow, J., Boye, J., Carlson, R., Edlund, J., Granström, B., House, D., and Wirén, M. (2000). AdApt—A Multimodal Conversational Dialogue System in an Apartment Domain. In *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, volume 2, pages 134–137, Beijing, China.

Jelinek, F., Lafferty, J., Magerman, D. M., Mercer, R., and Roukos, S. (1994). Decision Tree Parsing Using a Hidden Derivation Model. In *Proceedings of Darpa Speech and Natural Language Workshop*, pages 272–277.

Johnston, M. (1998). Unification-Based Multimodal Parsing. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL)*, pages 624–630, Montreal, Quebec, Canada.

Johnston, M. and Bangalore, S. (2000). Finite-State Multimodal Parsing and Understanding. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING)*, pages 369–375, Saarbrücken, Germany.

Johnston, M., Bangalore, S., Visireddy, G., Stent, A., Ehlen, P., Walker, M., Whittaker, S., and Maloor, P. (2002). MATCH: An Architecture for Multimodal Dialog Systems. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 376–383, Philadelphia, USA.

Kehler, A. (2000). Cognitive Status and Form of Reference in Multimodal Human-Computer Interaction. In *Proceedings of the 17th National Conference on Artifical Intelligence (AAAI)*, pages 685–689, Austin, Texas, USA.

Koons, D. B., Sparrell, C. J., and Thorisson, K. R. (1993). Integrating Simultaneous Input from Speech, Gaze, and Hand Gestures. In Maybury, M., editor, *Intelligent Multimedia Interfaces*, pages 257–276. Menlo Park, CA: MIT Press.

Lambert, L. and Carberry, S. (1992). Modeling Negotiation Subdialogues. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 193–200, Newark, Delaware, USA.

Litman, D. J. and Allen, J. F. (1987). A Plan Recognition Model for Subdialogues in Conversations. *Cognitive Science*, 11:163–200.

Neal, J. G. and Shapiro, S. C. (1988). Architectures for Intelligent Interfaces: Elements and Prototypes. In Sullivan, J. and Tyler, S., editors, *Intelligent User Interfaces*, pages 69–91. Addison-Wesley.

Neal, J. G., Thielman, C. Y., Dobes, Z., Haller, S. M., and Shapiro, S. C. (1998). Natural Language with Integrated Deictic and Graphic Gestures. In Maybury, M. and Wahlster, W., editors, *Intelligent User Interfaces*, pages 38–52. Morgan Kaufmann.

Oviatt, S. L. (2000). Multimodal System Processing in Mobile Environments. In *Proceedings of the Thirteenth Annual ACM Symposium on User Interface Software Technology (UIST)*, pages 21–30. New York: ACM Press.

Stent, A., Dowding, J., Gawron, J. M., Bratt, E. O., and Moore, R. (1999). The Commandtalk Spoken Dialog System. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 183–190, College Park, Maryland, USA.

Vo, M. T. and Wood, C. (1996). Building an Application Framework for Speech and Pen Input Integration in Multimodal Learning Interfaces. In *Proceedings of IEEE International Conference of Acoustic, Speech and Signal Processing*, volume 6, pages 3545–3548, Atlanta, USA.

Wahlster, W. (1998). User and Discourse Models for Multimodal Communication. In Maybury, M. and Wahlster, W., editors, *Intelligent User Interfaces*, pages 359–370. Morgan Kaufmann.

Wahlster, W. (2000). Mobile Speech-to-Speech Translation of Spontaneous Dialogs: An Overview of the Final Verbmobil System. In *Verbmobil: Foundations of Speech-to-Speech Translation*, pages 3–21. Springer Press.

Wu, L., Oviatt, S., and Cohen, P. (1999). Multimodal Integration - A Statistical View. *IEEE Transactions on Multimedia*, 1(4):334–341.

Zancanaro, M., Stock, O., and Strapparava, C. (1997). Multimodal Interaction for Information Access: Exploiting Cohesion. *Computational Intelligence*, 13(4):439–464.

Zhou, M. X. and Pan, S. (2001). Automated Authoring of Coherent Multimedia Discourse in Conversation Systems. In *Proceedings of the Ninth ACM International Conference on Multimedia*, pages 555–559, Ottawa, Ontario, Canada.