Operations for Context-based Multimodal Interpretation in Conversational Systems

Joyce Chai
IBM T. J. Watson Research Center
19 Skyline Drive
Hawthorne, NY 10532, USA
{jchai}@us.ibm.com

ABSTRACT
In a multimodal conversation, user inputs are usually abbreviated or imprecise. Only fusing inputs together is inadequate in reaching a full understanding. To address this problem, we have developed a context-based approach for multimodal interpretation. In particular, we present three operations: ordering, covering, and aggregation. Using feature structures that represent intention and attention identified from user inputs and the overall conversation, these operations provide a mechanism to combine multimodal fusion and context-based inference. These operations allow our system to process a variety of user multimodal inputs including those incomplete and ambiguous ones.

1. INTRODUCTION
To aid users in their information-seeking process, we are building an intelligent infrastructure, called Responsive Information Architect (RIA), which can engage users in a full-fledged multimodal conversation. Users can interact with RIA through multiple modalities (speech, text, and gesture), and RIA can act/react through automated multimedia generation (speech and graphics). Currently, RIA is embodied in a testbed, called Real Hunter™, a real-estate application for helping users find residential properties. As a part of this effort, we are developing a semantics-based multimodal interpretation framework MIND (Multimodal Interpreter for Natural Dialog) for interpreting multimodal user inputs.

Traditional multimodal interpretation has been focused on multimodal integration with limited consideration on the interaction context [2, 5, 8]. However, in a conversation, user inputs are usually abbreviated or imprecise, and the overall meanings of those inputs can only be identified based on various contexts. Therefore, we have developed a context-based interpretation approach. In this paper, we describe the use of conversation context to enhance multimodal interpretation. Conversation context has been investigated extensively in spoken dialog systems [1, 9]. Here we extend it to multimodal conversation.

Table 1 shows a conversation fragment between a user and RIA. The user starts by pointing to Briarcliff on the screen and requesting houses (U1), and RIA replies with available houses (R1). Then the user points to the screen and asks about the size information (U2). Since the user is not pointing precisely at a particular object (as shown in Figure 1), she could potentially refer to the nearby house or the town of Briarcliff. However, because houses are the focus of attention in the prior conversation, MIND interprets that most likely the user is asking about the size of the nearby house. Next, the user asks about the price (U3). Note that the user does not explicitly specify what object she likes to know the price for. Once again, based on the conversation context, MIND understands that the user is asking about the price of the same house as in U2. In the next turn (U4), the user does not specify the object of interest, and once again, the pointing is ambiguous. Nevertheless, based on the prior conversation, MIND is able to figure out that the user is most likely interested in both the size and the price of another house.

2. REPRESENTATION
To capture salient information identified from user inputs and the overall conversation, MIND uses feature structures [3] to represent intention and attention [6]. Intention describes the purpose of a conversation. Since we focus on information-seeking applications, we currently distinguish three top-level purposes: DataPresentation, DataAnalysis (e.g., comparison), and ExceptionHandling (e.g., disambiguation). These purposes are used as the types of intention feature structures. In addition, intention feature structures have two features: Act (a user...
For example, the intention of \( U_1 \) (Figure 2a) is represented by a DataPresentation feature structure that indicates the user is requesting RIA to identify (Met: identify) some information (show houses). The actual information to be presented (i.e., attention) is captured by the House feature structure that states the user is interested in a collection of houses located in the city of Briarcliff. Note that not all predefined features are included in each feature structure. Only those that can be instantiated are included. For example, the attention feature structure for \( U_1 \) does not have the Focus, Aspect and Content features since those are not specified in the input. Furthermore, since the purpose of an input (intention) and the semantic category of the content (attention) sometimes cannot be determined from user inputs directly, we have added a type Unknown. For example, in Figure 2(b), the attention structure of the Unknown type indicates that the semantic category of the content is not known (since the user did not specify the object of interest).

Using feature structures, we also model the conversation context in a conversation history based on Grosz and Sidner's conversation theory (1986). Figure 3 depicts the conversation history that outlines the first four turns of the conversation in Table 1. Specifically, our conversation history has three main elements: modality units, conversation units and conversation segments. A modality unit keeps track of intention and attention either recognized from each unimodal user input (\( U_1 \) speech input and \( U_1 \) gesture input in Figure 3) or specified by RIA for speech or graphics generation. A conversation unit records user (rectangle \( U_1-2 \)) or RIA (rectangle \( R_1-2 \)) overall meanings at a single turn in a conversation. These units are grouped together to form a conversation segment (oval) based on their intention and attention features. In case these elements, we use the same set of feature structures to consistently represent intention and attention. We show next that such consistency provides a computational basis for inference based on conversation context. It is worth mentioning that, in addition to intention and attention feature structures, each of the three elements also contain other information. For example, in the modality unit, a time stamp is recorded for each speech input and gesture input. Furthermore, a conversation history also maintains different relations between segments and between turns. Details can be found in [4]. In this paper, we focus on intention and attention.

### 3. MULTIMODAL UNDERSTANDING

MIND has three major processes: unimodal understanding, multimodal understanding and discourse interpretation. Unimodal understanding identifies semantic information from each unimodal input. Multimodal understanding combines multimodal inputs together to form an overall meaning. Discourse interpretation captures the overall progress of the conversation [4]. In this paper, we focus on multimodal understanding, which consists of two sub-processes: multimodal fusion and context based inference.

#### 3.1. Multimodal Fusion

During multimodal fusion, MIND first uses temporal constraints to align intention/attention feature structures identified from each modality, and then 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 [3]. Two feature structures can be unified if they have compatible types based on an inheritance hierarchy, and the values of the same features are also consistent. Otherwise, unification fails. Based on this nature, unification is applied in multimodal fusion since information from different modalities tends to complement each other [7]. However, in some cases, information provided by individual modalities might serve or reinforce each other. For example, the purpose of the gesture input in \( U_2 \) is referring and the purpose of the speech input is describing. Since the value of the Method feature for the gesture input (Met: Refer, Figure 4b) and the corresponding value (Met: Describe, Figure 4a) for the speech input do not satisfy any subsumption relationship, they cannot be unified through traditional unification operation. However, in a broader sense, these two purposes are not inconsistent. The referring purposes is more likely to serve the main purpose of describing. Therefore, in order to make those two feature structures usable, we have developed a new operation: ordering (\( \succ \)). To better illustrate ordering (and other operations), we use the following notations:

\( \hat{\text{Type A}} \) and \( \hat{\text{Type B}} \) are compatible if they hold subsumption relationship based on an inheritance hierarchy. By our default, an Unknown type subsumes all other known types.
Ordering combines two feature structures based on pre-defined partial orders (\(\succ\))\(^1\). As in Figure 5, if two feature structures are not compatible, then ordering fails (step 1). If values of a feature in both structures are themselves feature structures, then those values need to be combined through ordering again (recursion in step 2b). If values of common features in both structures satisfy some partial order, then those two structures can be combined (step 2c). Otherwise, ordering fails (step 2d). Furthermore, all features in two compatible feature structures are fused in the final combined structure (step 3, 4). For example, based on the partial order: \(\text{Met} \succ \text{Des}\), current partial orders are usually used for combining intention feature structures. The only difference between ordering and traditional unification operation is the addition of partial orders. Ordering offers flexibility since the definition of information being consistent is broadened through partial orders. Thus, by updating partial orders, MIND can customize different requirements for different applications and combine different information together.

However, ordering is not capable of resolving ambiguities. For example, in U2, since the Unknown type subsumes all other types, the Unknown attention feature structure in the speech input (Figure 4a) can be unified with either the House or the City feature structure for the gesture input (Figure 4b, there are two feature structures for attention because of the user's imprecise pointing). Thus, an ambiguity arises since two combined feature structures are created (Figure 4c-d). In this case, MIND uses the conversation context to resolve the ambiguity, which we will describe next.

### 3.2. Context-based Inference

Inference based on conversation context is supported by two operations: covering and aggregation.

#### Covering

In many cases, after multimodal fusion, the overall meaning of a user input still cannot be identified. For example, the exact focus of attention for U3 is not clear after multimodal fusion (Figure 6a). Therefore, the interpreter needs to use the conversation context (represented in the conversation segment DS2 in Figure 6b) to infer the information that is not explicitly specified in the user input. However, combining DS2 and U3 through unification/ordering would fail since the Aspect feature in both structures have different values (Size versus Price) that do not satisfy any partial order. Therefore, to make this kind of context useful in interpretation, we have defined another operation: covering (\(\overset{\circ}{\bowtie}\)).

Ordering A and B (\(A \overset{\circ}{\bowtie} B\)) creates \(C = [c_1: w_1, \ldots, c_n: w_n]\), in following steps:

1. If \(t_p\) and \(t_q\) are not compatible
   - then ordering fails.
2. else assign the more specific type to \(t_p\).
   - then ordering fails;  
3. else.
   - then ordering fails.
4. else if (\(t_p\) is a feature in both structures) then ordering fails.
   - then ordering fails.
5. else if (\(t_p\) is a feature in both structures) then ordering fails.
6. else if (\(t_p\) is a feature in both structures) then ordering fails.
7. else if (\(t_p\) is a feature in both structures) then ordering fails.

Figure 5. Ordering Operation

\(^1\) A subsumption relation can be viewed as a type of a partial order

Covering combines two feature structures by overlaying one structure (covering structure) on top of the other one (covered structure), and the values (for the common features) in the covering structure prevail (see Figure 7, where A is the covering structure and B is the covered structure). Specifically, if the types of two feature structures are not compatible, then the covering fails (step 1). For features in both structures, if they have inconsistent values and their values do not form any partial order, then the values from the covering structure always prevail and are included in the resulting structure (step 2c-iii) and 2d). Note that covering is also recursive (step 2b). Unlike ordering, covering is not commutative (i.e., \(A \overset{\circ}{\bowtie} B \neq B \overset{\circ}{\bowtie} A\)). Thus to perform this operation correctly, the direction is important. For example, to interpret U3, MIND applies covering U3 on DS2 (U3 \(\overset{\circ}{\bowtie}\) DS2). As a result, features in DS2 (Topic and Content) are added in the combined structure and the value of the Aspect feature is set to Price (Figure 6c). Thus, MIND is able to figure out that the user is interested in the price of the same house MLS2034765 (Multiple Listing Service ID) as in U2.

Conversation context also helps to resolve ambiguities resulting from multimodal fusion. Based on a heuristic that users tend to stay focused unless they explicitly or implicitly specify otherwise, MIND uses the most recent conversation segment to pick the most likely interpretation when an ambiguity arises. For example, just as in U1, because of the imprecise gesture input in U4, multimodal fusion creates two attention structures as in Figure 8(c). However, when unifying the City attention structure with the context, covering fails since the types of attention structures in Figure 8(c1) and DS2 (Figure 8d) are not compatible. Therefore, only the House structure in Figure 8(cii) can be unified with DS2 through covering. The resulting structure shows that in U4, the user is asking about the price and the size information about a different house (MLS2043876).

While maintaining explicitly specified information, covering offers a mechanism to systematically use the conversation context to infer covering.
Aggregation. In covering, the information explicitly specified in the prior conversation. However, in many cases, the old information should be used as an addition to the new information. For example, in U5, the user only requests for houses with the victorian style. Although the user did not mention the location, she is most likely requesting for victorian houses in Briarcliff, not anywhere else. Covering is still insufficient to support this type of context-based inference. Thus, we have defined another operation: aggregation (\(\bowtie\)).

Aggregation combines two feature structures by adding the values of features in one feature structure (supplement structure) to the values of the same features in the other structure (main structure) (see Figure 9, where B is the supplement structure and A is the main structure). There are two differences between aggregation and covering. The first one is that, instead of overwriting inconsistent values as covering does, aggregation simply adds those values in the supplement structure to the values of the same features in the main structure (step 4 is gone). Similar to covering, aggregation is also not commutative. For example, in interpreting U6 (Figure 10), MIND uses aggregation of DS1 onto U5 (U5 \(\bowtie\) DS1). Specially, MIND adds the value of the Constraint feature in the House feature structure in DS1 to the value of the Constraint feature in U5. Furthermore, since U5 is the main structure, the feature value in DS1 is not carried over to U5, and thus the Content in U5 still needs to be determined based on the revised constraints.

Aggregation uses the context to put additional constraints or specifications on an user input. This is particularly useful for information-seeking applications. Users may provide new information at each conversation turn. However, MIND should consider all requirements identified through the entire conversation when helping users with navigation.

As a summary, during the multimodal understanding process, MIND first performs multimodal fusion using ordering, then uses covering and aggregation to combine feature structures from the inputs with relevant conversation segments through instance-based learning.

4. CONCLUSIONS

To enhance multimodal interpretation in a conversation setting, we have developed a context-based approach in our interpretation framework MIND. In particular, we have developed three operations to extend the unification operation: ordering, covering and aggregation. Based on our feature structures that consistently represent meanings of user inputs and the overall conversation, these operations provide a mechanism to combine multimodal fusion and context-based inference in a unified manner. These operations allow MIND to process abbreviated and ambiguous inputs.

5. ACKNOWLEDGEMENTS

The author would like to thank Keith Houck, Shimei Pan, Rosario Uceda-Sosa and Michelle Zhou for their work on RIA and MIND.

6. REFERENCES