 Semantic representations in spoken dialogue systems

GABRIEL SKANZTE

gabriel@speech.kth.se

Department of Speech, Music and Hearing, KTH

In this paper, the requirements on semantic representations in spoken dialogue systems are discussed. It is argued that spoken dialogue needs a special treatment when it comes to natural language processing, since the phenomena that occur in spoken language are not the same as those that occur in written language. These special properties of spoken language are outlined and the different aspects that need to be represented are discussed. The most common semantic formalisms – first-order logic, feature structures and simple frames – are also presented and discussed with respect to these requirements. It is argued that spoken dialogue not just puts further requirements on semantic representations, but that the focus instead should shift away from problems that are mostly common in written language to those that are common in spoken language.

1 INTRODUCTION

In order to build dialogue systems, i.e. computer systems that can engage in a natural language dialogue, the user’s utterances must be given some sort of meaning. In other words, we must somehow make computational representations of them, to be able to perform meaningful actions, whether it is a database lookup, a feedback expression, or just a greeting. The complexity of these representations could vary, depending on what kind of phenomena we want to be able to handle. In this paper, we will focus on spoken dialogue. Spoken language is different compared to written language, and the discussions on semantics must take into account what kind of language use we are talking about.

This paper will only focus on what semantic representations we should strive for, not on how they could be derived. Thus, we will not look at parsing or compositional semantics, or any other semantic analysis process. I believe that it might be important to start in this end, in order to know what we want to accomplish with the semantic analysis.

We will start to discuss what makes spoken dialogue special when it comes to semantic representations. Spoken language is associated with a lot of uncertainty, fragmentary input and several modalities, that have impact on what semantic representations are needed. I will then discuss what aspects that need to be represented. There are many aspects of a single utterance that we need to take into consideration, but since the use of utterances in spoken dialogue is heavily context dependent, we must also have some representation of the context in which they take place. This also means that we must deal with how the semantics of an utterance relates to the semantics of the context. To do this, we will look at the most common semantic formalisms that have been used: first-order logic, feature structures and simple frames. We will then discuss to what extent these formalisms can handle the different phenomena that are typical for spoken dialogue.

2 WHAT MAKES SPOKEN DIALOGUE SPECIAL?

Using Chomsky’s terminology, one could say that the semantics of spoken dialogue deals with the performance of language. According to Poesio (1996), formal semantics (at least Montagovian-style) have instead been dealing with the competence of language, using grammatically perfect utterance as examples. This “performance” of language is associated with a lot of properties that needs to be discussed.
2.1 Uncertainty in spoken dialogue

One of the major problems with the semantics of natural languages is that of uncertainty and ambiguity. The issue of semantic representations is a general problem in the field of natural language processing, which has to be handled in most NLP applications. However, spoken dialogues seem to have some properties that make it special when it comes to semantic representations.

According to Heisterkamp et al. (1992) dialogue semantics has to deal with three sources of uncertainty. The first is the general ambiguity of language. Each word or sentence can often have several different meanings. The interpretation of an utterance also relies on the knowledge of the agent. A second problem inherent in dialogue is that the interpretation of an utterance often is heavily dependent on context. Anaphora, ellipses and deictic expressions are common in dialogue, as the speakers are trying to reduce the effort in forming their utterances and instead rely on the common ground shared with the other speaker. The interpretation of such expressions is dependent on the textual or non-textual context. A third source of uncertainty is specific for spoken dialogue systems. The textual utterance that is to be interpreted is the result of an error-prone speech recognition process. This means that the resulting representation may contain errors that are independent of the parsing algorithms. The problem here is not only how to make the correct interpretation, but also how to represent the uncertainty that comes from the noisy input. The dialogue manager must deal with hypothetical interpretations that may be proved to be wrong.

2.2 Fragmentary input

In spoken dialogue, utterances must not take the form of complete sentences. Instead, utterances often consist of fragments, filled pauses (like “ehm”), pauses, etc, which all may have meaning in the context. This somewhat unpredictable behaviour makes spoken language relatively ungrammatical, compared to written language. Poesio (1996) calls these units micro conversational events. In order to make adequate interpretations of utterances, such events should be handled as well, not just bee seen as noise in the input. Often, such utterances have interactive functions, which we will discuss later on.

Utterances can also be fragmentary due to disfluences. For example, one could make a self-correction in the middle of an utterance, such as “I want an apartment on Vasaväg... Valhallvägen”. Topicalizations are also common, such as “The green apartment ... what does it cost?” (Bell et al., 2001). Silent pauses in the middle of an utterance pose a problem. Is the utterance complete, or should the system wait for more input? This means that we should consider some sort of incremental parsing, i.e. interpret the utterance during the production. One could also consider letting the system take actions during the production, such as eliciting feedback or making interruptions. The problem of fragmentary input could be seen as just a parsing problem (which we will not face in this paper), however it is also important to be able to represent fragmentary input, if the system is supposed to take actions in the middle of an utterance. Bell et al. (2001) argues that utterances should be incrementally parsed and represented as closing or non-closing. If an utterance is determined as non-closing, the system should wait for more input, but it should also signal continued attention to the user by for example using facial gestures such as lowering the forehead, looking at the user. Since the utterances are incrementally parsed, there is also need for representation of fragments, which can be unified with the representation of later utterances.

2.3 Modalities in spoken dialogue

The semantic representations in spoken dialogue should not only reflect the textual content of utterances. People also use prosodic features and body gestures to convey meaning. Dialogue systems that are not only built to analyse or present textual information, are called multimodal. Multimodality puts further demands on semantic representations, since non-textual information is not very easy to represent in propositional terms. There is also the problem of modality integration. How should a gesture of negative attitude be unified with the propositional representation of an assertion? However, multimodality do not just put further demands on the dialogue system, multimodal utterances can also be used for mutual disambiguation of utterances, since the different modalities can express redundant informa-
tion (Oviatt, 1999a). Multimodal expressions can also replace complex verbal descriptions in spatial domains (Oviatt, 1999b). These topics are very important, however I will not discuss them further in this paper.

3 WHAT SHOULD BE REPRESENTED?

In dialogue, we have to represent both utterances and the context in which they appear, in order to relate the semantics of the utterance to the semantics of the context. We will start by looking at the representation of an utterance and then discuss the representation of context.

3.1 Representing utterances

A single sentence could often be represented as a proposition, expressing some sort of fact. However, this is not enough for representing the meaning of utterances in dialogue. First, utterances are most often spoken in order to perform some action other than just stating a fact. Second, all utterances seem to have double meaning, both task related meaning and conversational regulatory meaning. One must also consider how the uncertainty of the meaning of utterances should be represented.

3.1.1 Proposition and dialogue act

A sentence in natural language is often represented in semantics as some sort of proposition, where some predicates express properties of objects and/or relations between objects. Take for example the following utterance:

*The large apartment is nice*

In first-order logic, this could be represented as something like:

$$\exists x (\text{apartment}(x) \land \text{large}(x) \land \text{nice}(x))$$

We could also represent it with key-value pairs:

- **object:** apartment
- **properties:** [large, nice]

While these propositional representations seem to be adequate, the sentence – uttered in a conversation – probably has some more meaning. The speaker did not just want to express this as a fact, but instead accomplish something, maybe indirectly requesting an opinion from the other speaker, or trying to convince the other speaker to buy the apartment. The philosopher John Austin recognized that contributions in dialogue should be seen as performing actions apart from just uttering words – so-called *speech acts*. They can be seen as the intention of the utterance – what the user tries to achieve when it is uttered. Speech acts have been popular in dialogue systems as a way of classifying utterances. The original classifications of speech acts have been expanded and more common terms are now *dialogue acts* or *conversational moves* (Jurafsky & Martin, 2000). The recognition and representation of such dialogue acts has been a central issue in most dialogue systems, in order to perform the right actions. A question requires an answer; an assertion may require some sort of feedback.

Dialogue acts have been used for dialogue management in mainly two different ways; the *grammar-based approach* and the *plan-based approach*. In the former approach, the dialogue is described as a sequence of dialogue acts, and a finite state grammar for possible dialogues are created. The user’s utterance can then be interpreted in this context, and the dialogue act be determined. The grammar also prescribes what actions the system should take. In the plan-based approach, the dialogue acts are seen as indicating a plan that the user has. The system has to recognize this plan and reason about the user’s goals and intentions, in order to generate new plans so that the right actions can be taken.

3.1.2 Interactive function

Allwood et al. (1993) distinguish between main message functions and interactive functions of utterances. Main message functions are those that are concerned with solving the task at hand, e.g. requesting and asserting information. Interactive functions are concerned with things like turn taking and
feedback. Those functions are extremely important for the speakers, since they are used for handling how utterances should relate to context and what should be added to the common ground. According to Clark (1996) these kinds of functions should be seen as occurring in parallel with the main functions. A single utterance can for example contribute to the task, while at the same time display how something that was previously said was understood:

A: I would like an apartment with three rooms in this area.
B: Ok. There are five apartments in this area.

From a main message perspective, the underlined parts of B’s answer could be seen as irrelevant, since they do not provide any new information. However, they have the interactive functions of signalling how B understood A’s question.

According to Allwood et al. (1993), the first thing that needs to be represented from feedback is which levels of communication the feedback relates to. Communication occur at several levels:

- To be able to say and hear an utterance, the speaker and hearer must have contact.
- In order to understand an utterance, the hearer must perceive the utterance.
- The hearer must also understand the utterance.
- In order for the utterance to have the intended meaning, the hearer must also have a positive attitude towards the proposed dialogue act.

The feedback that the speaker gets on these different levels will provide evidence of how the utterance was interpreted by the hearer. According to the principle of downward evidence (Clark, 1996), evidence at a higher level is also evidence at a lower level. For example, if the speaker gets feedback on a positive attitude (which is in line with the intention), he also knows that there is positive contact, perception and understanding.

### 3.1.3 Representing uncertainty

In spoken dialogue, speakers do not just state facts that are automatically added to the common ground. Instead, the speakers negotiate on what is to be taken as common ground (Clark, 1996). One of the important interactive functions of utterances is to communicate understanding and non-understanding. Compared to human-human dialogue, the speech recognition process in spoken dialogue systems is much more error-prone. Therefore, the system should represent uncertainty in some way.

Heisterkamp & McGlashan (1996) describes a way of representing how the interpretation is derived and how it relates to context. They argue that the dialogue act should not be represented, since the dialogue act represents a user intention, which the dialogue system knows very little about. Instead, the propositional content should be assigned a contextual function, which can be one of the following:

- new_for_system(X)
- repeated_by_user(X)
- inferred_by_system(X)
- modified_by_user(X)
- negated_by_user(X)

The contextual function of the utterance determines what dialogue goals are created, so that the right actions can be taken, such as confirmations, etc.

### 3.2 Representing context

Spoken dialogue can be an extremely efficient medium for informational exchange. Ginzburg (1996) mentions two properties of dialogue that make this possible: discursive potential and ellipsis. The first
property means that the topics that a speaker can select for discussion, at a given point, are extremely limited. This, in turn, allows the use of ellipsis; short utterances and fragments that have meaning in the specific context. These properties will also put demands on the dialogue system in representing the context that allows them.

The (textual) context can be seen as the sum of the contributions that have been jointly added to the common ground – the mutual knowledge that is shared between the participants (Clark, 1996). The question is what this context looks like and how it changes after each utterance.

3.2.1 Ginzburg’s model of context semantics

In the “classical” view, as described by Ginzburg (1996), context changes as participants make assertions. The propositional content of the assertion is added to the common ground, if none of the other participants object. Thus, the context is the sum of propositions added to the common ground. According to Ginzburg, this model is too simplified, and he extends it with what he calls update semantics:

- Contributions are not just accepted or rejected; there is also room for discussion. According to Clark (1996), the process by which a contribution is taken as common for the participants, involves phases of presentation and acceptance. This process, called grounding, makes it possible for the participants to ensure that they form the same interpretations of what has been said. Speaker A presents something for speaker B to accept. In order to ensure that this is mutually understood, A needs some signal of understanding. These signals can be of different strengths, depending on the grounding criterion, i.e. the consequences of misunderstanding and task failure. The signal of understanding can range from a simple “uhu” to a more explicit “I think you said Stockholm”.

- The context should be updated with illocutionary information, i.e. what dialogue act is expressed. The main distinction is made between assertions and queries. Queries raise questions for discussion, which are put in a partially ordered repository called QUESTIONS UNDER DISCUSSION (QUD). For example, if question q is maximal in QUD, it is permissible to provide information about q or a question q₁ on which q depends.

- One important feature of discourse is locality, or adjacency. This means that an utterance relates in a special way to the preceding utterance. The LATEST MOVE should be represented, in order to make an accurate interpretation of the next utterance. It is permissible to make moves that are available as reactions to the latest-move, but not ones that directly relate to something that was said several turns ago, without raising it as a new question.

3.2.2 Discourse history and structure

Ginzburg describes how facts and queries relate to the common ground. However, there are other aspects that also should be represented. To make the conversation efficient, the speakers coordinate their use and interpretation of referring expressions. In order to reduce the amount of negotiation, the speakers will strive to establish a common ground. This common ground, built up between speakers, will not only consist of common beliefs, but also of common means of referring to those beliefs, resulting in a coordination of referring expressions. As shown by Garrod and Andersson (1987) in experiments on human-human conversation, speakers build conceptual pacts during the dialogue, which are specific for the discourse shared between them. Thus, in order to interpret the referring expressions that the user will use, the system should keep track of what has been referred to in the previous discourse, but also which referring expressions were used.

This representation of previous discourse is often called discourse history. The history might not only contain past referents, but also the structure of discourse. In Grosz & Sidner’s (1986) model of discourse, the central aspects that need to be accounted for are the discourse segments (building up the discourse structure) and the focus stack (keeping track of past referents). This model has influenced the development of the dialogue system LINLIN (developed at Linköping University), where the discourse history is represented as a tree with the utterances as nodes (Jönsson, 1993).
3.2.3 Non-textual context

The context that has been discussed in this section is only the textual context that models of dialogue often are limited to. One should also consider the non-textual context. One obvious case is when we use deictic expressions, such as “this” or “that”. These are often accompanied with pointing gestures that need to be interpreted as well. We can also talk about a wider context, such as the situation where the interaction takes place, and the specific user of the dialogue system. This has ramifications for user adaptation in dialogue systems. Pakucs (2002) describes a system where semantic frames (see section 4.3) can be used in representing adaptation to user preferences and situation context. The system learns how the user usually fills in the form, depending on time and location.

4 SEMANTIC FORMALISMS

To summarize, a semantic formalism for spoken dialogue systems should allow us to:

- Represent the utterance adequately (propositional content, dialogue act, interactive function, uncertainty).
- Represent the context and relate the semantics of the utterance to the semantics of the context.
- Allow us to use the representation in order to execute some action in the application, or extract some information from a knowledge source.

In this section, I will present some formalisms that are common in dialogue systems. The examples will be very basic and not deal with all of the problems mentioned above. However, they should give a picture of whether they are feasible for doing this or not.

4.1 First-order logic

In first-order logic (or FOL), the world consists of objects, properties and relations. FOL has been a common way of representing knowledge in AI, for example in building knowledge bases. A knowledge base consists of a set of propositional facts, expressed in FOL. This seems to be a convenient way of representing both knowledge and context in dialogue. If utterances are represented in FOL, we can relate them to the context and background knowledge, in order to make inferences. This has been utilized in plan-based approaches to dialogue (Jurafsky & Martin, 2001).

In the planning paradigm, we can define a dialogue act as an action with preconditions, effect and body. The precondition states what facts need to be true in order for the action to apply. The Effect tells what will be true after the action is performed. The body states how the knowledge base should be updated if the action is executed.

For example, a request can be represented as

\[
\text{Request} \left( \text{Speaker, Hearer, Act} \right):
\begin{align*}
\text{Precondition:} & \quad \text{Want} \left( \text{Speaker, Act(Hearer)} \right) \\
\text{Effect:} & \quad \text{Want} \left( \text{Hearer, Act(Hearer)} \right) \\
\text{Body:} & \quad \text{Believe} \left( \text{Hearer, Want} \left( \text{Speaker, Act(Hearer)} \right) \right)
\end{align*}
\]

Using these action definitions, the system can reason about what actions should be performed, but also recognize what plans the user might have. The recognition of a dialogue act is thus seen as a plan recognition problem.

Using FOL, the system can reason about the user’s intentions. As we have seen, intentions are represented as predicates, such as Belief, Know and Want. We could then define for example Know as

\[
\text{Know}(A, P) \leftrightarrow P \land \text{Believe}(A, P)
\]

Since the context can be represented as a set of logical facts, we can also define rules for incorporating utterances with context.
4.2 Feature structures

A feature structures (or attribute-value matrix) is a set of feature-value pairs where a value can either be an atomic value or another nested feature structure. Central to feature structures is that they can be unified with each other. This has proved to be useful in constraint-based grammars, where morphosyntactic features from different nodes in a parse tree have to fit together. Unification ensures that, for example, a plural noun phrase has to be followed by a plural verb phrase.

Feature structures have also been used for semantic representations. I will take the dialogue system LiNLIn (Jönsson, 1993), developed at Linköping University, as an example. The dialogue act and other information about the utterance (for example the speaker) are stored as feature-value pairs at each node in the dialogue tree. The dialogue tree is a tree-structure which presents the past dialogue moves and therefore also the textual context. The propositional content is represented as feature-value pairs like this:

**What is the price of Honda Civic?**

```
Object:  
  [Manuf: Honda]
  [Model: Civic]
Properties:  
  [Aspect: Price ]
```

The system response to such a question is presented as a table with the cars matching the query. Now, the system use unification in order to make contextual interpretations of subsequent utterances. For example, suppose the next question from the user is “How large is the boot of the 85?”, referring to the Hondo Civic 85 presented in the table previously. The representation of this utterance is unified with the feature structure in the previous node in the dialogue tree, yielding the representation:

**How large is the boot of the 85?**

```
Object:  
  [Manuf: Honda]
  [Model: Civic]
[Year: 1985]
Properties:  
  [Aspect: Price ]
  [Aspect: Boot ]
```

Unlike FOL, the mapping between the semantic formalisms and the structure of the knowledge base is often less straightforward. According to Jurafsky & Martin (2001), feature structures can be translated into equivalent FOL. A common solution is, however, to write some sort of transformation rules to translate the feature structures to SQL queries or similar. The feature structure should then correspond to the query language used at the application level. In LiNLIn, for example, a semantic representation has a slot called *objects* and a slot called *properties*. The objects represent rows in a database table, and the properties represent columns, so that constructions of SQL statements are easily done:

```
Objects:  
  [Manuf: Honda]
  [Model: Civic]
Properties:  
  [Aspect: Price ]
```

```
SELECT price FROM bilar WHERE manuf="Honda" AND model="Civic"
```

4.3 Frames

The simplest form of representation is the so-called *frame* or *slot-filler*, utilized in many commercial dialogue systems. According to Jurafsky & Martin (2001), there is no difference between feature structures and frames, but we will for convenience use the notion of frames to refer to non-nested feature structures, with simple key-value pairs. These kinds of frames are used in VoiceXML; a dialogue
specification scheme developed by W3C for telephony based spoken dialogue applications (VoiceXML Forum, 2001). The advantage of those simple representations is that they are very easy to specify for the developer. The grammar can easily be augmented with semantic tags that assign values to the slots.

Which slots are filled can be seen as a simple representation of the local context. This allows so-called mixed-initiative dialogue, i.e. the user can fill several slots in one utterance or be prompted for each slot. For example, suppose a flight travel booking system needs information about departure city, destination city and departure time in order to find matching flights. The user could start by saying “I want to fly from Gothenburg to Stockholm”. These two slots are filled and the system finds out that one slot is missing, so it asks the question assigned to this slot: “What time do you want to travel?” Unification is normally not used in order to match the utterance semantics with context semantics, since the non-nested structure make no use of this. Instead, if a new value is given for a slot (if the user changes his mind or makes a correction) the old value is just replaced with the new.

5 CONCLUSIONS & DISCUSSION

The choice of semantic formalism seems to be associated with the kind of approach to dialogue that is taken. The plan-based approach, which needs a lot of reasoning, uses a logical formalism. In the grammar-based approach, it is instead more common to rely on the unification of feature structures at the nodes in the dialogue parse tree. Simple frames are used in systems that only need to handle one (simple or complex) query at a time, where the frame can represent the different sub-topics of the query.

Feature structures seem to be good for representing partial input (common in spoken language), which needs to be unified with other partial input. The predicate can for example be represented in one slot and the object in another, thus separating different parts of the propositional content. Simple frames are less good for this, since the whole proposition (most often) is represented in one slot. The handling of fragmentary input does not seem to be much discussed in systems based on FOL.

Representation of uncertainty needs more complex representations than just simple frames, since the propositional content must be separated from the representation of how this content was derived. Simple frames do not give an account for discourse history (at least not non-local history), and are more suitable for short dialogues with just a few turns.

Complex semantic relations seem to be hard to represent with feature structures, or impossible with simple frames. However, the question is to what extent these complex relations occur in spoken human-computer dialogue. It is important to note that spoken dialogue may not just add more things we must account for in semantic representations. Some phenomena of natural language that have been popular to study in formal semantics may not be the most important ones to handle in spoken dialogue systems. For example, it is hard to find any articles discussing the handling of ambiguous quantifier scope in spoken dialogue systems. This is probably due to the limited use of constructions that look like “two foreign languages are spoken by every linguist” in such systems. Most systems would not even be able to represent such complex meanings, even if they could decide upon an interpretation. The problem of pronoun resolution also seems to be a limited problem in dialogue systems, based on the number of articles that are published on this subject. The examples taken in most articles on formal semantics are very seldom taken from spoken language.

We have only looked at semantic representations, not how they can be derived. The fragmentary input in spoken dialogue also needs more robust parsing techniques. Deep parsing of spoken utterances seems to be hard, since the grammar of spoken language is very irregular.
6 REFERENCES


