User Models, Discourse Models, and Some Others

Karen Sparck Jones

Computer Laboratory
University of Cambridge
Cambridge, England

My context for this note is natural language interfaces to complex systems, considered, as the aim is system building, from the system's point of view. I hold that the discourse model relates four other models that are all required in dialog, but which need to be distinguished. These other models, as I define them in this note, are the world model, the user model, the system model, and the text model. It is a mistake, in my view, to try to establish the relation between discourse models and user models without reference to the other models that figure in the interaction between man and machine.

The case for all the models mentioned can be made through a hypothetical system example. Suppose we have an automated reference librarian in an industrial company, i.e., a highly sophisticated expert system. Suppose also that the system can sustain the kind of dialog we can, and so imagine a dialog between the system and a user as follows:

U1. I'm looking for a book on logic programming.

S1. You're in Accounts, aren't you?

U2. Yes.

S2. So you'd like something with financial examples.

U3. Well that would be very convenient, but I'm really interested in logic programming itself.

S3. Just as well: we don't have anything on Prolog for accountants. Do you want something really noddy? We've got one pretty basic book.

U4. How good do you think it is?

S4. It's very popular. It doesn't assume you know anything about programming.

U5. Oh, but I've done a bit of Cobol.

S5. There isn't anything on Prolog from a Cobol point of view, I'm afraid.

U6. I wouldn't have thought that mattered: Cobol's a good general lead in, isn't it?

S6. Not especially. But we have another book that would probably suit you better than the first: it's very clear and has a lot of different examples like planning the week's menus and building an office meeting diary system, but it does assume you have some experience of programming.

I assume that the system knows about the material in the library, about the company, and about the library's role in the company's activities. It also knows about the company's employees (which include the illustrative user, known to be a woman in Accounts who thinks her boss underestimates her). The system's knowledge of the library and the company are part of its permanent world knowledge, and its knowledge of the company's employees is in principle no different in kind, though the system's knowledge of people's intensional states may be less certain than some of its other knowledge.

There is nevertheless a good reason for distinguishing the system's knowledge of its users, and hence of any specific user, from the rest of its world knowledge. This is a functional reason: the system exists to solve problems presented by its users. The individual user is a necessary participant in the system's problem solving, and the system's actions are driven by its view of the user's specific needs. In the dialog, the system's response S6 is motivated by the perceived fact that the user, as a user, is a person with particular characteristics who is separated out from the rest of the world because her needs have to be met. Thus the response is an appropriate one in relation not to individual utterances or facts about the user, but to the entity constituting the user model as a whole, namely that she's a non-novice female, seeking to impress her boss by improving her computing skills. (This functional view, of course, implies that whether the user is human or not is an independent, contingent matter.)

But for the same functional reason, the system has to have a model of itself embodying, for example, its plan to extract more detail about the user's book request. Though the system is in principle, like the user, part of the world, it has to be functionally distinguished for it to carry out its task: thus it is the system's aim, not the general state of the world, which leads it to choose S2, asking about financial examples.

0362-613X/ 88 /0100●-●$03.00

However, there are no good grounds for assuming that the system's immediately accessible world model for a given interactive session contains all its world knowledge, its immediate user model everything it knows about the user, or its immediately accessible system model everything it knows about itself: for example, that its models for the illustrative dialog cover its knowledge of the company's buildings, the user's pay, or its own plan to revise its book descriptions (it's a powerful system). The system's entire stock of knowledge is not needed for effective interaction in a particular dialog: and indeed, given the large amounts of permanent knowledge presupposed, having all this to hand would simply clog everything up.

For a given dialog, therefore, the world model, user model, and system model will be selections, which are functionally motivated and hence distinguished, from the system's complete and previously undifferentiated stock of knowledge. This, of course, requires an invocation mechanism of the kind sought by Sperber and Wilson (1986), yet one which not merely selects information as relevant but assigns it a functional role. For example, being in Accounts becomes part of the user model, but knowledge about the relation between Prolog and logic programming becomes part of the world model, and proffering specific books as a strategy for clarifying user needs becomes part of the system model. The invocation is continuous, triggered by the progress of the dialog, so the system's knowledge of the user's desire to impress her boss, for example, which was not necessarily initially selected as relevant, is invoked to motivate S6.

The fourth factor is the dialog itself. The text of the interaction is an inert object: but discourse processing implies some model of the text as a linguistic object (Sparck Jones 1983). This linguistic model is a text model that is functionally required, like the other models, to support one subprocess of the whole problem-solving interaction. The text model deals with text-based entities, which are linguistically characterized and need not have real world referents, like the Cobol book (cf. Panel VII in Tinlap-3 1987). It is this linguistically motivated interpretation of the dialog text which embodies information, e.g., about the lexical items used, topicalisation, and anaphoric links. The text model is required to guide further discourse production, if not determining at least suggesting the choice of a word, constraining sentential structure to maintain cohesion, etc., as in the use of "accountants" rather than "financial staff" in S3, for example and of "that" in U6, and in the form of S5, where the structure of "Prolog from a Cobol point of view" is linguistically related to S3 and U5.

Compared with the other models, the text model may not appear to be a subset of a larger body of knowledge about the world, including knowledge about individuals. But it can apply general knowledge of argument structures (in the sense of Reichman (1985)) as well as of grammar, just as the permanent world model contains general knowledge about humans, and it can in principle also depend on prior knowledge of linguistic individuals, e.g., (somewhat trivially) "Accounts". Knowledge provided by the text model, like that supplied by the other models as they develop through the interactive session, can update the permanent stock, in which it has no more special status or character than the information about people who can function on occasion as users.

But the text model is not the discourse model: it is too shallow for this. It does not, for instance, express the relation between Prolog and logic programming, or the user's inferred intention to better herself, though these are clearly discourse matters. The text model includes, on the other hand, information about the order of mention of entities that is relevant to the production of linguistic responses, which is not obviously a discourse matter if the discourse model is about the substantive relations between discourse entities (Cohen, Morik, Schuster, all this issue).

In fact, none of the four models listed can claim to be a discourse model. The discourse model is what relates these four models, that is, expresses the relations between them. Thus the discourse model relates the world model entity, the noddy book, with the user model element representing the user's request for a book on logic programming; and it relates the system model constituent—help a user by suggesting a specific book—to these world model and user model entities, the noddy book and requested book, respectively. Again, the discourse model relates the system's belief about the utility of a book on Prolog for a would-be logic programmer with the world model link between Prolog and logic programming, and it relates this book in the world model with the text model entity for "one pretty basic book".

The discourse entities are the entities involved in the various models (taking entities as complex structured objects as well as simple ones), but the discourse model is not the mere aggregate of the other four models: it has to explicitly relate their elements in a way that allows transitions from one model to another to meet the requirements of the system's task. Thus the user's belief about the connection between Cobol and logic programming is an element of the user model, where it is functionally associated (by the system) with user's goal of learning about logic programming. But the user's belief about the relation between Cobol and logic programming is also related to the world model because it is a function of the world model to test for existence, in this case for the reality of a connection between Cobol and logic programming. The relation between these user and world model entities referring to Cobol and logic programming, i.e., the relation between a belief in a user model functionally concerned with achieving goals and a proposition in a world model functionally concerned with existence testing, is one sort of discourse model relation. There is another one in the relation between the user model belief and the text model entity "a good general lead in". But what discourse model relations we should recognize, and hence whether we should have an austere or promiscuous ontology for these relations, is not so far clear.

Thus the four different models—world, user, system, and text—below the discourse model are all needed because each serves a distinct function, implying a different selection from, and organization of, the total information relevant to the interaction between the user and the system. The discourse model is then also needed to provide the links between the four, which support calls from one to another.

A shifting focus of attention like that represented by the point of interaction between participants and discourse in Grosz and Sidner's (1986) account is naturally presupposed here. But my argument is that we need to separate a participant's (in this case, the system's) view of itself from its views of the world and of the user. The way these interact with the text model will then be reflected in a subset of relations (and hence entities) in the discourse model which constitutes the focus of attention. Thus, within the type of framework Grosz and Sidner propose, I am advocating, on functional grounds, a finer allocation of entities to a larger set of models. This large set of functionally motivated models is needed for serious systems and interfaces, though in simple cases the different models may collapse into one. I am similarly arguing, in relation to Wahlster (1987), for a finer set of modeling distinctions. These distinctions reflect the different status of the various objects and perspectives involved in discourse, and have to be recognized even if the system's operations are treated as if carried out within one overall model.

References

Grosz, B. and Sidner, C. 1986 Attention, Intentions, and the Structure of Discourse. In *Computational Linguistics* 12: 175–204.

Reichman, R. 1985 *Getting Computers to Talk Like You and Me*. MIT Press, Cambridge, MA.

Sparck Jones, K. 1983 Shifting Meaning Representations. In Proceedings of the 8th International Joint Conference on Artificial Intelligence: 621–623.

Sperber, D. and Wilson, D. 1986 *Relevance*. Basil Blackwell. Oxford, England.

Tinlap-3, Panel VII on Reference 1987 In Wilks, Y. (ed.), *Theoretical Issues in Natural Language Processing*, New Mexico State University, University Park, NM: 128–154.