# Dialog Modeling Within Intelligent Agent Modeling

**Marjorie McShane and Sergei Nirenburg**

Department of Computer Science and Electrical Engineering
University of Maryland Baltimore County
1000 Hilltop Circle, Baltimore, MD 21136
{marge,sergei}@umbc.edu

**Abstract**

In this paper we explore to what extent the modeling of dialog can be subsumed under the broader modeling of intelligent agents. Specifically, we analyze how four core aspects of modeling human physiology and general cognition can be applied to dialog, and attempt to determine whether dialog poses any modeling challenges that require additional methods or expressive power.

## Introduction

Novel applications can challenge traditional conceptions of how best to tackle a problem. Sometimes this happens more than once. A case in point: in the late 1970s Perrault and Allen (see, e.g., Allen and Perrault 1980) suggested a goal- and plan-based approach to dialog processing, influenced by classical AI approaches to planning and the work of Cohen, Levesque and Perrault (e.g., Cohen 1978, Cohen and Perrault 1979, Cohen and Levesque 1980). Later work in dialog processing (cf., e.g., Larsson et al. 2002, Lemon and Gruenstein 2004) has come to rely predominantly on dialog cues. This latter approach has moved away from (a) integration with a general goal- and plan-based conception of agency and (b) applying goals and plans specifically to dialog processing. This choice of this approach was justified in terms of feasibility understood in at least two different but complementary ways. The first – pessimism with respect to massive knowledge acquisition – has quickly propagated from its origins in expert systems to natural language processing and has strongly contributed to the paradigmatic shift toward knowledge-lean methodologies, stochastic ones chief among them. Second, building comprehensive models of intelligent agents was considered to be beyond the scope of a single research team; therefore, general problem solving and dialog processing were adjudged to have a better chance of success through the collaboration of teams working on these two issues.

As a result of these two feasibility-oriented concerns, a cornerstone of dialog processing has been its reliance on dialog models crafted for, and solely dedicated to, dialog. This is true even of DIPPER (Bos et al. 2003), a dialog system architecture that (a) claims to combine the strengths of the goal-oriented and cue-oriented paradigms implementing the information-state approach (Traum and Andersen 1999) and (b) uses "aspects of dialogue state as well as the potential to include detailed semantic representations and notions of obligation, commitment, beliefs and plans" (Bos et al. 2003).

From a purely scientific point of view, participation in dialog is just one of many capabilities inherent in a cognitively complex intelligent agent. It is natural, therefore, that with the growing prominence of work on naturally inspired intelligent agents that emulate human performance over a broad spectrum of everyday and specialized tasks, the separation of dialog processing from general problem solving capabilities has started to look increasingly less justified. Indeed, even from the standpoint of efficiency, integrating the two functionalities promises to bring about the benefits of economies of scale.

We have been exploring issues related to incorporating dialog into a comprehensive intelligent agent at our current stage of work on Maryland Virtual Patient (MVP)[1], an agent-oriented simulation and mentoring environment aimed at automating certain facets of medical education and assessment. The agent network in the system is composed of both human and artificial agents. The human agents include the user (typically, a trainee) discharging the duties of an attending physician and, optionally, a human mentor. The software agents that are built to simulate human behavior include the virtual patient, lab technicians, specialist consultants and a mentoring agent. The system also includes an array of non-humanlike software agents.

The core agent is the virtual patient (VP), a knowledge-based model and simulation of a person suffering from one or more diseases (e.g., Jarrell et al. 2007, 2008; McShane et al. 2007a,b). The virtual patient is a "double agent" in that it models and simulates both the physiological and the cognitive functionality of a human. Physiologically, it undergoes both normal and pathological processes in re-



[1] Patent pending.

sponse to internal and external stimuli. Cognitively, it experiences symptoms, has medical and lifestyle preferences (a model of character traits), is capable of remembering, forgetting and learning, makes decisions, and communicates with the trainee about its personal history, symptoms and preferences for treatment. Users can interview a VP; order lab tests; receive the results of lab tests from technician agents; receive interpretations of lab tests from consulting physician agents; posit hypotheses, clinical diagnoses and definitive diagnoses; prescribe treatments; follow-up after those treatments to judge their efficacy; follow a patient's condition over an extended period of time; and, if desired, receive mentoring from the automatic mentor.

Work on MVP has progressed from the simpler (although not simple!) development of realistic, interactive physiological simulations of a half dozen diseases to the development of the cognitive functioning of VPs, which includes interoception (the experiencing of internally generated stimuli, like symptoms), decision-making, memory management, natural language processing, and the ability to learn lexical and ontological information. All of these capabilities have been modeled and implemented in a demonstration version of the system. The current system does not, however, include a dedicated dialog model. As we progress toward enhancing the VP's – as well as the mentor's – language capabilities, we are reviewing what have been considered to be central components of dialog models and the extent to which they can or cannot be subsumed under our generalized cognitive architecture. In other words, can we generalize without losing precision and compromising on depth of coverage? In this paper, we examine some theoretical and practical issues involved in making this judgment, focusing on four aspects of cognitive modeling that have application both inside and outside of the realm of dialog: (1) meaning representation, (2) the use of remembered goals, plans and scripts, (3) decision-making, and (4) learning.

## Meaning Representation

The representation of meaning is central to semantically-oriented text processing. When working with meaning, many NLP practitioners find it appropriate to use elements of a metalanguage and surface strings in conjunction: e.g., case roles (e.g., Gildea and Jurafsky 2002) or template slots (e.g., Hobbs et al. 1997)) might be filled by strings. Others consider it necessary to represent meaning using a metalanguage independent of any natural language. This requires that language strings be interpreted and converted into and out of that metalanguage. While the latter approach is typically considered more expensive, since semantic analysis and the component disambiguation are notoriously difficult, it offers clear benefits to an NLP system, permitting it to do its main work over unambiguous meaning representations. (See Bar Hillel 1970 for a discussion of the division between NLP and reasoning.) Moreover, if an intelligent agent has functionalities beyond language processing (as is assumed, e.g., also in Allen et al. 2001), the use of the same metalanguage for the representation of both linguistic and non-linguistic meaning offers even greater benefits, as evidenced by our experience with MVP.

In MVP, all physiological, general cognitive and language processing capabilities of all agents rely on the same ontological substrate, the same organization of the fact repository (agent memory of assertions) and the same approach to knowledge representation. This approach, which was originally developed for the knowledge-based semantic processing of language – and, in fact, has been used in the OntoSem semantic analyzer and its predecessors for almost two decades (see, e.g., Nirenburg and Raskin 2004; Beale et al. 1997) – has been seamlessly applied to the representation of all meaning in MVP, from physiological simulation to all aspects of agent knowledge, memory and reasoning. So, we seem to have a constructive proof that not only is a special type of meaning representation for dialog not necessary, it would be detrimental to overall agent functionality since it would require the agent to treat information from different sources – in MVP, interoception, language input and the results of the agent's own reasoning – in different ways, which is neither efficient nor conceptually plausible. It is noteworthy that basically no extensions to the expressive power of the OntoSem ontology or metalanguage have been required to support this new application: indeed, there is no fundamental difference between a script that permits a language processing agent to interpret a restaurant scenario (to choose an example at random) and a script describing the progression of a disease that permits the operation of a simulation. Although space does not permit a detailed discussion of the OntoSem and MVP environments, select observations and examples should make clear the benefits of a unified approach to meaning representation for a multi-functional intelligent agent.

Our meaning representations (MRs) are composed of ontological concepts and their instances, linked by ontologically defined properties, using the OntoSem ontology as the substrate. The interpretation and generation of language invokes an English lexicon whose entries are described with reference to ontological concepts, either directly or with property-based modifications (McShane et al. 2005a). When MRs convey the meaning of text we call them TMRs (text meaning representations) – a topic we have written about extensively in the past. Consider the following example, which shows various aspects of MR creation and use in MVP:

1. The physiological disease simulation creates an instance of pain in the VP at a certain time and with a certain intensity. Let us call this instance, which is remembered by the system as a trace, MR-1. This pain is experienced via (simulated) interoception by the VP. The VP's MR, MR-2, may or may not look like MR-1 due to the fact that the VP's ontology is not expected to match that of the "expert" ontology that underpins the simulation. For example, the

simulation might invoke ontological concepts (e.g., PERISTALSIS) that are known to a physician but not to a lay person. As a result of this mismatch, the system must carry out ontological paraphrase to support VP interoception, permitting the VP to interpret and remember the symptom in its own terms (for further discussion, see McShane et al. 2008a,b).

2. If the user asks the VP about pain, the VP interprets the English string, creating a TMR – we'll call it MR-3 – whose elements ontologically match those of the VP's memory. (For paraphrase in this context, see McShane 2008a,b.) The VP records MR-3 in its memory, as part of its dialog record, and uses its content to look up the appropriate response in its memory.

3. When the VP has found the answer, it generates a TMR, MR-4, that acts as input to the English language generator. In other words, this TMR constitutes the result of the content specification step in generation.

4. When the automatic mentor observes and records the user-VP conversation, it uses the MRs generated and interpreted by the VP, not needing to independently process English strings. By contrast, when the mentor converses with the user, it carries out the same kind of language processing and memory management as does the VP.

In sum, all of the knowledge stored and manipulated in MVP is represented using the same metalanguage, with conversion to and from English needed only for VP-user interactions and mentor-user interactions.

As an aside, there actually is one other use for metalanguage-to-English conversion in MVP: it is used to show the VP's thinking processes as part of the under-the-hood view of the simulation, where the knowledge structures being produced throughout the simulation can be viewed dynamically. Generating an English "window" into the workings of the system is, of course, not a native part of the system; however, this is useful when explaining the system to observers, since the formal metalanguage structures cannot be easily read and interpreted. To take one example: say the VP has been seen by the doctor, who may or may not intervene, and has been told to come back for a follow-up in 9 months. However, at 7 months the VP experiences a spike in symptoms. Whether or not the VP will decide to present early – and if so, when it will present – depends, in the current implementation, upon the actual level of the symptom(s), the VP's symptom threshold (how severe symptoms have to be in order to take action), the amount of time left before the scheduled appointment, and the degree of the sudden change in symptoms. (Of course, one could add any number of other factors, such as how much the VP likes or dislikes going to the doctor, whether the VP has time to see the doctor, how nervous the VP is about its health, etc. These are planned for the future.) At a given point in time the VP's thoughts might be: *My symptoms have increased significantly. They were mild when I saw the doctor last and now they are moderate. I have a high tolerance for symptoms. My appointment is 2 months away. I will wait until my appointment.* Of course, this type of evaluation is carried out regularly, so the VP might still decide to present early. Each of these "thoughts" is automatically generated from the functions that comprise the VP's decision-making process.

## Scripts, Plans and Goals

Dialog models must include mechanisms that permit the conversational agent to know what to do next – if a question is posed the agent should answer it, if the interlocutor clearly didn't understand an utterance, the agent should clarify (this is an aspect of "grounding", as discussed, e.g., in Traum 1994), and so on. Allen and Perrault (1980) introduced goal- and plan-based reasoning in the study of dialog interpretation. Later frameworks preferred to rely on the notion of discourse obligation (e.g., Traum and Allen 1994). That is, the interlocutor's utterance sets up a discourse obligation on the part of the agent, and the agent must fulfill this obligation in the way specified by an associated function. The hypothesis we have been investigating is that agent behavior in a dialog can be modeled using the same goal-driven methods used to initiate all other agent action in MVP, without the need for dialog-specific obligations. To show why we think this is possible and preferable – at least for our multi-faceted agents – we must start with an overview of the use of goals, plans and scripts in MVP.

As mentioned earlier, the MVP simulation employs ontologically recorded complex chains of events. We distinguish two types based on agency and the extent to which the events are goal-driven.

**Scripts** in our approach are unagentive complex events for which positing goals would be a stretch, since the goals would need to be attributed to non-sentient, questionably sentient (e.g., bacteria) or divine sources: a heart beats (in order to keep the human in which it resides alive); a disease progresses (for whatever goals the instruments of disease – e.g., bacteria – fulfill by perpetuating the disease); food passes from the esophagus to the stomach (so that it can nourish the eater). Let us consider the example of a disease script more closely. In MVP, diseases are modeled as processes (non-humanlike agents) that cause changes in key property values of a VP over time. For each disease, a set number of conceptual stages is established and typical values or ranges of values for each property are associated with each stage. Relevant property values at the start and end of each stage are recorded explicitly, while values for times between stage boundaries are interpolated. The interpolation currently uses a linear function, though other functions could as easily be employed. A disease model includes a combination of fixed and variable features. For example, although the number of conceptual stages for a given disease is fixed,

the duration of each stage is variable. Similarly, although the values for some physiological properties undergo fixed changes across patients, the values for other physiological properties are variable across patients, within a specified range. The combination of fixed and variable features represents, we believe, the golden mean for disease modeling. On the one hand, each disease model is sufficiently constrained so that patients suffering from the disease show appropriate physiological manifestations of it. On the other hand, each disease model is sufficiently flexible to permit individual patients to differ in clinically relevant ways, as selected by patient authors. (See Jarrell 2007, 2008 and McShane 2007a,b for detailed descriptions of disease models.)

**Plans**, by contrast, are agentive and are used as a means of satisfying some agent's **goal**: going to the doctor and buying over-the-counter medication are two plans for the goal of healing illness; eating and taking diet pills are two plans for the goal of satisfying hunger; accepting the doctor's recommendation for a medical test and inquiring about other diagnostic options are two plans for fulfilling the goal of diagnosing disease. Goals are states, and states are formally represented in the OntoSem framework as objects with specific property values. For example, the goal of being healthy, held by a particular person, is represented as (human-x (health-attribute 1)), where 1 signifies the highest value on the abstract scale [0,1].

What follows is an informal sketch of the agent's manipulation of goals and plans in the current version of the MVP environment. The main goal pursued by all VPs in our environment is BE-HEALTHY. We assume that this is a universal goal of all humans and, in cases in which it seems that a person is not fulfilling this goal – e.g., a person makes himself ill in order to be cared for by medical professionals, or a patient selects bad lifestyle habits that damage his health – he is simply prioritizing some other goal, like BE-FOCUS-OF-ATTENTION or EXPERIENCE-PLEASURE, over BE-HEALTHY. In MVP, when a VP begins to detect symptoms, the goal BE-HEALTHY is put on the goal and plan agenda. It remains on the agenda and is re-evaluated when: (a) its intensity or frequency (depending on the symptom) reaches a certain level; (b) a new symptom arises; or (c) a certain amount of time has passed since the patient's last evaluation of its current state of health, given that the patient has an ongoing or recurring symptom or set of symptoms: e.g., "I've had this mild symptom for too long, I should see the doctor." At each evaluation of its state of health, the VP can either do nothing or go to see the doctor – a decision that is made based on an inventory of VP character traits, the current and recent disease state and, if applicable, previous doctor's orders (cf. next section). If it decides to see the doctor, that plan is put on the agenda. All subgoals toward achieving the goal BE-HEALTHY and their associated plans are put on and taken off the agenda based on VP decision functions that are triggered by changes in its physical and mental states throughout the simulation. So when the doctor suggests having a test (goal: HAVE-DIAGNOSIS) and the patient agrees, having the test (a plan toward the above goal) is put on the agenda; and so on.

Returning to dialog modeling, we are attempting to determine whether our plan- and goal-based methods are sufficient to support all the needs of dialog. Although we have not yet explored all of the issues involved, preliminary indications are that they very well may be.

Consider again the use of "obligations" in dialog modeling. Obligations have been used as the explanation for why, e.g., a participant in a dialog must respond to a question even if the answer is essentially vacuous, such as, "I don't wish to respond" (Traum and Allen 1994). However, obligations can be recast in terms of plans and goals: speakers can have the goal of being a polite member of society (which has its own benefits), which in turn has a series of conditional plans: if an interlocutor asks a question, answer it; if the interlocutor has misunderstood, help him or her to understand. These are the same sorts of rules as are found in the obligation-oriented models but their theoretical status is different. The goal BE-POLITE does not exclusively apply to verbal actions in dialogs, it applies as well to physical actions, e.g., not slamming the door in the face of a person entering a building behind you. So, if an intelligent agent – like our VP – is endowed with action capabilities beyond the realm of dialog, generalizations about its overall goals in life should be incorporated rather than splitting up its behavior into dialog-related and non-dialog-related categories.

Another aspect of many dialog models is an agent's understanding of the appropriate interpretation of utterances as dictated by the speech context. For example, when the doctor asks "How are you?" the agent should respond differently than if a colleague had asked the same question. Situation-based disambiguation of this sort can be carried out with the use of ontologically recorded plans. Specifically, an agent's GO-TO-DOCTOR plan encodes its knowledge of what can typically happen at a doctor's visit: greeting, small talk, asking about health and family history, doing a physical exam, positing a hypothesis or diagnosis, discussing treatment options, and so on. Upon receiving a language input, the agent must attempt to match the input to one of the expected points in the plan; if successful, this guides the selection of a response. Use of the same plans can help to detect if the interlocutor has misunderstood an utterance by the agent. In short, any deviation from the expected plan is a clue to the agent that it might need to take repair action (see Traum et al. 1999).

One aspect of dialog modeling that we believe might require a special approach is dialog-specific language conventions, such as: the resolution of personal pronouns; full interpretation of fragments or ellipsis (depending on how one chooses to linguistically analyze structures like "How often?", whose action must be recovered from the previous utterance); semantic ellipsis, which we define as the non-expression of syntactically non-obligatory but semantically obligatory material; etc. (See McShane et al. 2004, 2005b, McShane 2005 for OntoSem approaches to these phenom-

ena.) The approaches to some of these issues will apply to text as well as dialog.

## Decision-Making

All cognitive architectures are essentially grounded in a perception – decision – action loop, and the decision-making of all intelligent agents relies on decision functions that take parameter values as input and return a decision as output. Painted in these broad strokes, decision-making in MVP is quite traditional. However, one feature distinguishes the decision-making of VPs from that of most other intelligent agents: for VPs, character traits are central input parameters. The need for distinguishing character traits in creating a large, realistic, highly differentiated population of VPs is undeniable – after all, some patients are fearful, others distrust doctors, still others believe they know more than the physicians they consult… and all of these traits directly impact the patient's behavior in the medical context. However, the utility of modeling character traits is not limited to decision-making about life actions, it extends to the realm of dialog as well: some agents should be talkative and others reticent; some should provide more information than asked for and others should respond in monosyllables; some should use technical terminology and others should use laymen's terms; and so on. Endowing artificial agents with linguistically-oriented traits that permit them to be as distinguished in their mode of communication as they are in other aspects of their lives will, we hypothesize, enhance the suspension of disbelief that is key to an interactive application like MVP.

We have already seen one example of decision-making, using the example of deciding when to present to the doctor. Here we look at a different example in a bit more detail (for more in-depth discussion of decision-making in MVP, see Nirenburg et al. 2008b). Among the decisions a patient must make is whether or not to agree to a test or procedure suggested by the doctor, since many interventions involve some degree of pain, risk, side-effects or general unpleasantness. Some patients have such high levels of trust, suggestibility and courage that they will agree to anything the doctor says without question. All other patients must decide if they have sufficient information about the intervention to make a decision and, once they have enough information, they must decide whether they want to (a) accept the doctor's advice, (b) ask about other options, or (c) reject the doctor's advice. A simplified version of the algorithm for making this decision – the actual decision tree is too detailed to be included here – is as follows.

1. IF a function of the patient's trust, suggestibility and courage is above a threshold OR the risk associated with the intervention is below a threshold (e.g., in the case of a blood test)
   THEN it agrees to intervention right away.

2. ELSE IF the patient feels it knows enough about the risks, side-effects and unpleasantness of the intervention (as a result of evaluating the function enough-info-to-evaluate)
   AND a call to the function evaluate-intervention establishes that the above risks are acceptable
   THEN the patient agrees to the intervention.

3. ELSE IF the patient feels it knows enough about the risks, side-effects and unpleasantness of the intervention
   AND a call to the function evaluate-intervention establishes that the above risks are not acceptable
   THEN the patient asks about other options
   IF there are other options
   THEN the physician proposes them and control is switched to Step 2.
   ELSE the patient refuses the intervention.

4. ELSE IF the patient does not feel it knows enough about the intervention (as a result of evaluating the function enough-info-to- evaluate)
   THEN the patient asks for information about the specific properties that interest it, based on its character traits: e.g., a cowardly patient will ask about risks, side effects and unpleasantness, whereas a brave but sickly person might only ask about side effects.
   IF a call to the function evaluate-intervention establishes that the above risks are acceptable
   THEN the patient agrees to the intervention.
   ELSE the patient asks about other options
   IF there are other options
   THEN the physician proposes them and control is switched to Step 2.
   ELSE the patient refuses the intervention.

The two decision functions called by this function are presented in Nirenburg et al. 2008b.

Not all human-like agents in MVP need be endowed with character traits. For example, the mentor can be completely devoid of personality, both linguistically and non-linguistically, as long as it effectively assists the user as needed. This does not mean that MVP mentors will be undifferentiated – quite the opposite. We have already implemented mentoring settings that make the mentor provide more or less explanatory information and make it intervene at more or fewer types of junctures. In addition, we plan to add to our current mentoring model additional models that reflect the differing clinical beliefs and preferences of different experts. (It should be noted that much of clinical knowledge is derived from the experience of individual physicians, and that experience can differ greatly across physicians, leading to differing, though potentially coexisting, mental models.) However, these differences lie outside the realm of character traits.

Although we have not yet fully incorporated the influence of character traits on dialog behavior into our nascent dialog processing, a simplified distinction was implemented in an earlier demonstration version of MVP. There, VPs were categorized as medically savvy or medically naïve, with the former providing more information than asked for by the user and the latter providing only what the user explicitly asked for, thus requiring the user to ask follow-up questions.

## Learning

Within the field of NLP, machine learning can involve (among many other approaches) learning by reading (e.g., Forbus et al. 2007) and learning by being told. The latter idea ascends to McCarthy 1958 and was developed in systems such as Teiresias (Davis 1982) and Klaus (Haas and Hendrix 1983). Being able to rely on the OntoSem text understander, our VP uses a richer and less constrained channel of communication between the teacher and the learner than earlier systems, though, not surprisingly, the quality of text analysis is far from perfect. Still, the VP can already learn by being told (Nirenburg et al. 2008a), and work is underway (e.g., Nirenburg and Oates 2007) on having it learn by reading. However, its learning does not derive from language alone – the VP can learn from its simulated experiences as well. Whether the learning is based on language or non-linguistic experience, the modeling strategy is the same, as is the effect on agent knowledge.

Three VP knowledge bases are augmented during a simulation: the ontology – information about the world in general, the fact repository ("memory of assertions") – facts about instances of events in the world, and the lexicon – the form and meaning of lexical strings. We have already discussed augmentation of the fact repository; here we briefly describe our approach to learning ontology and lexicon.

The VP can learn ontology and lexicon through discourse with the trainee (learning by being told) or by reading texts, e.g., those found on the web (learning by reading). When the VP encounters a new word, it creates a new lexical entry for it with as much meaning as is immediately available. For example, if the trainee says "Your test results reveal that you have achalasia," and if the VP has never heard of achalasia, it can hypothesize – based on its GO-TO-DOCTOR plan – that achalasia is some sort of a disease (we will not nitpick as to the difference between a disease, disorder, etc.). Thus, the lexical entry "achalasia" will be created, mapped to the ontological concept DISEASE. If the doctor provides more information about the disease, the VP uses these descriptions to fill the associated property slots in its DISEASE concept. Of course, all of the text processing and ontology and lexicon population use the metalanguage referred to above. If we have a curious patient (we do not yet, but we will), then that patient can, between doctor visits, search for information on the web to fill out its ontological specification of its disease or any other poorly understood concepts and return to the doctor/trainee for clarifications, questions, and so on. Thus, dialog-based and reading-based learning can co-occur in the VPs of MVP.

As concerns ontology, apart from learning it through language input, VPs can learn it from direct experience. For example, say a VP agrees to a procedure whose pain level it thinks will be acceptable, but during the procedure the VP realizes that the pain level is far greater than it can tolerate; when proposed that procedure again, the VP can decline based on its revised interpretation of the value of "pain" for the procedure.

## Discussion

MVP utilizes knowledge-rich approaches to NLP and agent modeling that were more widely pursued 20 or 30 years ago than they are today. Interest in plan- and goal-based R&D, as well as deep-semantic NLP, dwindled when investigators concluded that they were too labor-intensive to support practical applications. However, these conclusions must be put into perspective, particularly when juxtaposing past efforts with MVP.

**First**, most of the research on plan- and goal-based reasoning was devoted to creating systems that *developed* plans on the fly. In MVP, by contrast, we imposed the constraint that the system would not be required to develop plans, it would only be required to use preconstructed plans. This simplifying constraint is well-suited to MVP since system users will not be asked to solve never before seen types of cases, and the system itself – in the guise of the virtual mentor – will not be asked to invent novel approaches to patient care or fundamentally creative responses to questions. **Second**, in our environment various types of simplifications are possible with no loss in the quality of the final application. For example, we are not attempting to model every known aspect of human physiology, we are modeling only those that are needed to support a lifelike simulation at a grain-size that fulfills all foreseen teaching goals; we are not planning to supply our mentor with all possible mentoring moves (see, e.g., Evans and Michael 2006), only those sufficient to support the needs of medical students, whose primarily learning through MVP will, we hypothesize, derive from trial and error during practice simulations; and we are not attempting (at least not yet) to configure an open-domain conversational agent but, instead, one that converses well in the more constrained – but certainly not toy – domain of doctor-patient interviews, for which we can realistically develop sufficient ontological and lexical support. **Third**, MVP is a high-end application that requires sophisticated simulation, language processing and generalized reasoning capabilities that are not supported by the types of (primarily stochastic) methods that have been of late attracting the most attention in NLP.

In this paper we have discussed the results of our current, application-driven exploration of the possibility of subsuming a "dialog model" under a more generalized approach to agent modeling. Our goal in creating a unified modeling strategy, with little or no need for highly specialized components, is to create intelligent agents whose functionality can expand in any way – as by the addition of vision or haptics – and into any domain without the need for extensions to the base environment. In addition, we are trying to capture overarching generalizations about agent behavior like the one cited above: a human-like agent should respond to a question in dialog for the same reasons as it does not slam the door in the face of someone entering

a building behind it. We believe that such conceptual, modeling and implementational generalizations will lead to the development of agents that will not be disposed of every time a new capability is required of them. This, to our minds, is a pressing goal in building the next generation of intelligent agents.

## References


Allen, J., Ferguson, G., and Stent, A. 2001. An Architecture for More Realistic Conversational Systems. In *Proceedings of the Conference on Intelligent User Interfaces*, 1-8. January 14-17, Santa Fe, New Mexico.

Allen, J. F., and Perrault, C. R. 1980. Analyzing Intention in Dialogues. *Artificial Intelligence* 15:3, 143-178.

Bar Hillel, Y. 1970. *Aspects of Language*. Jerusalem: Magnes.

Beale, S., Nirenburg, S., and Mahesh, K. 1995. Semantic Analysis in the Mikrokosmos Machine Translation Project. In *Proceedings of the 2nd Symposium on Natural Language Processing*. Bangkok, Thailand.

Bos, J., Klein, E., Lemon, O., and Oka, T. 2003. DIPPER: Description and formalisation of an information-state update dialogue system architecture. In *Proceedings of the 4th SIGDial Workshop on Discourse and Dialogue*, Sapporo, Japan.

Cohen, P. 1978. On Knowing What to Say: Planning speech acts. Technical Report 118, Department of Computer Science, University of Toronto.

Cohen, P. R., and Levesque, H. J. 1980. Speech Acts and the Recognition of Shared Plans. 1980. In *Proceedings of the 3rd Conference of the Canadian Society for Computational Studies of Intelligence,* Victoria, B.C., pp. 263-271.

Cohen, P. and Perrault, C. R. 1979. Elements of a plan-based theory of speech acts. *Cognitive Science,* 3:177-212.

Davis, R. 1982. Teiresias: Applications of Meta-level Knowledge. In Davis, R., and Lenat, D., *Knowledge-Based Systems in Artificial Intelligence*. New York: McGraw-Hill.

Evens, M., and Michael, J. 2006. *One-on-One Tutoring by Humans and Computers*. New Jersey and London: Lawrence Erlbaum and Associates, Publishers.

Forbus, K., Riesbeck, C., Birnbaum, L., Livingston, K., Sharma, A., and Ureel, L. 2007. Integrating Natural Language Knowledge Representation and Reasoning, and Analogical processing to Learn by Reading. In *Proceedings of AAAI-07.*

Gildea, D., and Jurafsky, D. 2002. Automatic Labeling of Semantic Roles. *Computational Linguistics*, 28(3): 245-288.

Haas, N., and Hendrix, G. 1983. Learning by Being Told: Acquiring knowledge for information management. In Michalski, R., Carbonell, J., and Mitchell, T. (eds.), *Machine Learning: An AI Approach*. Palo Alto, CA: Tioga.

Hobbs, J., Appelt, D., Bear, J., Israel, D., Kameyama, M., Stickel, M., and Tyson, M. 1997. FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text. *Finite-State Language Processing*, MIT Press, Cambridge, MA.

Jarrell, B., Nirenburg, S., McShane, M., Fantry, G., Beale, S., Mallott, D., and Razcek, J. 2007. An Interactive, Cognitive Simulation of Gastroesophageal Reflux Disease. *Studies in Health Technology Informatics,* 194 –199. IOS Press.

Jarrell, B., Nirenburg, S., McShane, M., Fantry, G., and Beale, S. 2008. Revealing the Conceptual Substrate of Biomedical Cognitive Models to the Wider Community. *Studies in Health Technology Informatics*, 281 – 286. IOS Press.

Larsson, S., Berman, A., Grönqvist, L., and Kronlid, F. 2002. TrindiKit 3.0 Manual. SIRIDUS deliverable D6.4.

Lemon, O., and Gruenstein, A. 2004. Multithreaded Context for Robust Conversational Interfaces: Context-sensitive speech recognition and interpretation of corrective fragments. *ACM Transactions on Computer-Human Interaction*, 11(3): 241-267.

McCarthy, J. 1958. Programs with Common Sense. In *Proceedings of the Symposium on the Mechanization of Thought Processes*. Teddington, UK.

McShane, M., Beale, S., and Nirenburg, S. 2004. OntoSem Methods for Processing Semantic Ellipsis. In *Proceedings of the HLT/NAACL 2004 Workshop on Computational Lexical Semantics*, Boston, MA.

McShane, M., Nirenburg, S., and Beale, S. 2005a. An NLP Lexicon as a Largely Language Independent Resource. *Machine Translation* 19(2): 139-173.

McShane, M., Nirenburg, S., and Beale, S. 2005b. Semantics-Based Resolution of Fragments and Underspecified Structures. *Traitement Automatique des Langues* 46(1): 163-18

McShane, M., Nirenburg, S., Beale, S., Jarrell, B., and Fantry, G. 2007a. Knowledge-Based Modeling and Simulation of Diseases with Highly Differentiated Clinical Manifestations. In *Proceedings of the 11th Conference on Artificial Intelligence in Medicine (AIME 07)*. Amsterdam, The Netherlands, July 7-11.

McShane, M., Fantry, G., Beale, S., Nirenburg, S., Jarrell, B. 2007b. Disease Interaction in Cognitive Simulations for Medical Training. In *Proceedings of the MODSIM World Conference, Medical Track*, Virginia Beach, Sept. 11-13.

McShane, M., Nirenburg, S., and Beale, S. 2008a. Resolving Paraphrases to Support Modeling Language Perception in an Intelligent Agent. In *Proceedings of the Symposium on Semantics in Systems for Text Processing (STEP 2008)*, Venice, Italy.

McShane, M., Nirenburg, S., and Beale, S. 2008b. Two Kinds of Paraphrase in Modeling Embodied Cognitive Agents. In *Proceedings of the Workshop on Biologically Inspired Cognitive Architectures*, AAAI 2008 Fall Symposium, Washington, D.C., Nov. 7-9.

Nirenburg, S., McShane, M., and Beale, S. 2008a. A Simulated Physiological/Cognitive "Double Agent". In *Proceedings of the Workshop on Naturally Inspired Cognitive Architectures*, AAAI 2008 Fall Symposium, Washington, D.C., Nov. 7-9.

Nirenburg, S., McShane, M., Beale, S., and Jarrell, B. 2008b. Adaptivity in a Multi-Agent Clinical Simulation System. In *Proceedings of AKRR'08, International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning*. Porvoo, Finland, Sept. 17 – 19.

Nirenburg, S., and Oates, T. 2007. Learning by Reading by Learning to Read. In *Proceedings of the International Conference on Semantic Computing*. Irvine, CA, August.

Nirenburg, S., and Raskin, V. 2004. *Ontological Semantics*. Cambridge, MA: MIT Press.

Traum, D. 1994. A Computational Theory of Grounding in Natural Language Conversation. PhD thesis, Department of Computer Science, University of Rochester, Rochester, NY.

Traum, D. R., and Andersen, C. F. 1999. Representations of Dialogue State for Domain and Task Independent Meta-Dialogue.

In *Proceedings of the IJCAI'99 Workshop on Knowledge and Reasoning in Practical Dialogue Systems*.

Traum, D., Andersen, C., Chong, Y., Josyula, D., Okamoto, Y., Purang, K., O'Donovan-Anderson, M., and Perlis, D. 1999. Representations of Dialogue State for Domain and Task Independent Meta-Dialogue. *Electronic Transactions on Artificial Intelligence* 3(D):125-152.

Traum, D., and Allen, J. 1994. Discourse Obligations in Dialogue Processing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.