Resolving Paraphrases to Support Modeling Language Perception in an Intelligent Agent

Sergei Nirenburg
Marjorie McShane
Stephen Beale
Universtity of Maryland Baltimore County (USA)
email: `sergei@umbc.edu`

Abstract

When interacting with humans, intelligent agents must be able not only to understand natural language inputs but also to remember them and link their content with the contents of their memory of event and object instances. As inputs can come in a variety of forms, linking to memory can be successful only when paraphrasing relations are established between the meaning of new input and the content of the agent's memory. This paper discusses a variety of types of paraphrases relevant to this task and describes the way we implement this capability in a virtual patient application.

1 Overview of and Rationale for Studying Paraphrase

Paraphrase, under any of its many definitions, is ubiquitous in language use. It could be likened to reference, both in function and in the complexity of its detection and resolution. Indeed, there are many ways to express a given idea in language: one can use a canonical word/phrase (*dog*), a synonymous terse locution (*mutt, pooch, canine, man's best friend*), or an explanatory description that can be of any length and include one or more specific salient features (*a pet that barks*; *one of the two most common four-legged domesticated mammals in the USA that is not a cat*). Although these locutions are not semantically identical, they are *functionally* equivalent in many contexts, meaning that they can permit a person or intelligent agent to carry out the same types of reasoning.

No matter which of the above locutions is used to express the idea of *dog*, a person or an artificial intelligent agent should be able resolve it to the concept DOG in his/its world model. Such resolution, or "anchoring", permits other knowledge about the entity to be leveraged for reasoning: for example, the sentence *Our pooch has a long tail* should be construed as perfectly normal, whereas *Our pooch wrote a grocery list* should be understood as impossible in its direct sense since dogs cannot be agents of writing. Such incongruence should, in turn, suggest either a non-real world or the use of *pooch* as a nickname for some person or intelligent agent, like an automatic grocery list writing system.

1.1 Work by Others

Paraphrase is a difficult problem: at its deepest, it centrally involves semantics, which, due to its inherent complexity, can be addressed only in limited ways in current NLP work. As a result, most contributions devoted to paraphrase can be described as syntactic or "light semantic." In some contributions, processing semantics is constrained to finding synonyms, hyponyms, etc., in a manually constructed word net, like WordNet or any of its progeny. Some others do not rely on a manually constructed knowledge resource but, rather, aim to determine distributional clustering of similar words in corpora (see, e.g. Pereira et al. (1993) or Lin (2001)). A few approaches to dealing with paraphrase actually go beyond the detection and use of synonyms. For example, Lapata (2001) seeks to interpret the meanings of contextually elastic adjectives (such as *fast*, which means different things in *fast highway* and *fast eater*) by semi-automatically constructing paraphrases for phrases that include such adjectives. These paraphrases use the original noun and the adjective (or any of its synonyms, taken from a hand-constructed list) in its adverbial form and add a corpus-derived candidate verb intended to explain the meaning of the adjective. Results are evaluated by human judgments of whether a paraphrase (e.g., *highway travel quickly*) is appropriate as an explanation of the meaning of *fast* in *fast highway*.

Ibrahim et al. (2003) pursue the more immediate goal of supporting a question-answering system. Creating paraphrases for questions helps to expand the queries to the textual resources that are mined for answers. In an early version of this system, such paraphrase rules — which included a combination of lexical and syntactic transformations — were created by hand (Katz and Levin, 1988). The new approach follows the methodology of Lin and Pantel (2001) for dynamically determining paraphrases in a corpus by measuring the similarity of paths between nodes in syntactic dependency trees. This method was applied to pairs of sentences from different English translations of the same text. (The idea of using a monolingual "sentence-aligned" corpus is due to Barzilay and McKeown (2001).) Ibrahim et al. (2003) then suggest a set of heuristics for the subsentential-level matching of nouns and pronouns which leads to the specification of paraphrases in terms of rules such as *X became a state in Y* $\leftrightarrow$ *X was admitted to the Union in Y*. The reported precision of the process is about 41%, while the upper bound is given at about 65%.[1] Ibrahim et al. state that "question answering should be performed at the level of 'key relations' in addition to keywords." We believe that it is even better to use key *word senses* rather than key *words,* and to include key relations of a semantic and pragmatic nature — though syntactic information should be retained as a valuable source of heuristics for specifying semantic relations. We believe that we have developed enabling technologies and resources that allow us, at this time, to process paraphrase by relying on meaning representations rather than just syntactic dependencies and text-level relations.

One system that has an application area similar to ours is the one developed by Boonthum (2004). Boonthum is developing an automatic tutoring application that will be enhanced by paraphrase recognition. To process paraphrase, she automatically converts natural language sentences into Conceptual Graphs (Sowa, 1983) and compares the graphs of two candidate paraphrases using various metrics. This system, unlike ours, works at the level of strings (not concepts), does not automatically carry out disambiguation, and cannot handle complex sentences or long spans of text.

Since paraphrase recognition, when viewed broadly, is a very challenging task, some developers choose to focus on a narrow application area. One such system, reported in Brun and Hagège (2003), detects paraphrases in texts about toxic products. Developers hand create rules using lexical and structural information, and system output is logical structures like PHYS_FORM(acetone,liquid), which means that the physical form of acetone is liquid. The approach taken in this work seems very appropriate for this narrow domain of interest.

1.2 Our research methodology

The research methodology we adopt has the following features, which will serve to orient it in the landscape of work by others. This methodology:

- addresses paraphrase within an application;
- takes into account the needs of question answering and — more broadly speaking — dialog processing;
- integrates paraphrasing due to different types of agent perception: the perception of language and the perception of non-linguistic inputs, like interoception (sensitivity to stimuli originating in the body, e.g., symptoms of a disease);
- uses an agent's memories as both the source of paraphrase detection and as the target to which new memories are linked; and

[1] We believe that the low upper bound is due to the way the problem was framed. In cases where the semantic differences among candidate paraphrases are important (not "benign"), the inter-respondent agreement, we believe, will be higher.

- has provisions for including conceptual paraphrases, which are different ways of describing the same object or event that must be interpreted using the ontological knowledge available to specific agents.

The initial experimentation that we are reporting covers a relatively narrow domain but we hypothesize that the same methodology can be used in other domains, with certain modifications related to ontological and lexical coverage.

1.3 Maryland Virtual Patient (MVP)

The application that drives our current research is Maryland Virtual Patient (MVP), which is an agent-oriented simulation and tutoring system. In MVP, a human user plays the role of a physician in training who must diagnose and treat open-ended simulations of patients, with or without the help of a virtual mentor agent (e.g. McShane et al., 2007). The virtual patient is, itself, a "double" agent, comprised of: (a) a physiological agent that lives over time and responds in realistic ways to disease progression and interventions, and (b) a cognitive agent that experiences symptoms, decides when to consult a physician, makes decisions about its lifestyle, treatment options, etc., and communicates with a human user using natural language. The system currently covers six diseases of the esophagus, so many of our examples will come from this subdomain of medicine.

As should be clear even from this brief overview, MVP is a reasoning-intensive application. Both physiological simulation and NLP are supported by hand-crafted, ontologically grounded knowledge that includes:

1. a general purpose ontology with broad and deep coverage of medical concepts, including ontological scripts describing disease progression and treatment, the plans and goals of patients and physicians, clinical best practices, medical interviews and dialog in general

2. a lexicon whose entries include a syntactic structure, a semantic structure (linked to the ontology), and calls to procedural semantic routines (e.g., to provide for the reference resolution of pronouns and other deictics)

3. a fact repository, which is a memory of assertions, as contrasted with the ontology, which covers knowledge of types.

All knowledge in the MVP environment is recorded using the metalanguage of description of Ontological Semantics (Nirenburg and Raskin, 2004). The MVP application will serve as a concrete example for the discussion of paraphrase processing in applications that include intelligent agents. However, the analysis is readily generalizable and could be applied to any system that would benefit from paraphrase understanding.

This paper will not discuss all types of paraphrase and how OntoSem (the implementation of the theory of Ontological Semantics) handles them, even though one of the core contributions of OntoSem is the robust handling of lexical and syntactic paraphrase by automatically deriving identical meaning representations for inputs that contain such paraphrases (for discussion see Nirenburg and Raskin, 2004, Chapter 8). Here we focus on just a few of the more "compositional-semantic" types of paraphrase and our theoretical and implementation-oriented solutions to treating them.

2 Paraphrase-Oriented Eventualities

Each agent in MVP is supplied with its own ontology, lexicon and fact repository (i.e., memory), which can be enriched on the fly in various ways based on the agent's activities — be they linguistic, interoceptive, or other. In order for the language-endowed agents (on whom we focus here) to operate intelligently — as when answering questions posed by the human user or learning new facts he presents to them — they must be able to **interpret** language input, **remember** the content of that input, and attempt to **match/link** that content with memories already stored in their fact repository. Linking new information to old memories is a standing goal of all intelligent agents, and in MVP it is triggered automatically for each new input. A core capability enabling such linking is the recognition and resolution of paraphrase.

We will show how various types of paraphrase are handled as part of agent memory management in the OntoSem environment. More specifically, we focus on *creating* and *linking* new knowledge from linguistic input, not on the use of this knowledge for reasoning. Memory management (e.g., modeling forgetting and generalizing) is also a key enabling technology, but one whose description lies outside of the scope of this paper.

Having generated a meaning representation (MR) for a textual input, the intelligent agent must consider the following eventualities in deciding on how to remember the content of this input. The eventualities in boldface (numbers 5, 6 and 7) are those that we will be discussing in some detail below.

1. The newly input MR is identical to a stored memory

2. The newly input MR is identical to a stored memory except for metadata values: the identity of the speaker, the time stamp, etc. This can be viewed as type coreference. For example, in the MVP environment, if the agent coughs every day, is every cough a new instance or is it better remembered as a generalized action with a given periodicity? The answer here depends in a large part on the event generalization capabilities of an agent (a component of its memory management capabilities): indeed, even in real life one cannot be immune from the failure to realize that a certain sequence of events is actually better viewed as a single periodic event.

3. The newly input MR contains a subset or a superset of properties of a stored memory. For example, the new input can describe only the location of a symptom but not its severity, whereas remembered instances of this same symptom may overtly list its severity and various other properties. Note that the information about which properties are applicable to a particular concept is stored in the ontology; the memory (fact repository) contains information about those of the properties that were overtly perceived by the agent.

4. The new input is similar to a stored memory but one or more properties has a different value. For example, an input could specify one level of symptom severity while stored instances of the symptom may specify different values of this property.

5. **The newly input MR (or a component of it) is related to a stored memory via ontological subsumption, meronymy or location.**
6. **The newly input MR is related to a stored memory as the latter's precondition or effect.**
7. **The newly input MR is related to a stored memory via "ontological paraphrase."**
8. The new input is not related to any stored memory because different concepts are used, there are conflicting property values, etc. For example, a symptom experienced by somebody other than the given agent may be known to the agent, but the agent will certainly not interpret it as coreferential with knowledge about its own symptoms.

For case 1, the new information is interpreted as confirmation of the existing memory, it is not stored as a separate memory. For case 2, the choice of storing instances individually or grouping them into a recurring event is determined by the agent's memory management activities. For cases 3-8 reasoning must be carried out to determine if there is a match or not. In our current implementation, if a stored MR unifies with the newly input MR, the two MRs are judged to be paraphrases. With respect to case 4, this is a simplification because significant differences in values of properties in the two MRs under comparison should be used as heuristics voting against declaring the two MRs paraphrases. However, this level of analysis requires the establishment of a scale of relevancy on all the properties of a given concept, a task that we defer to future system releases. For case 8, the new information should be stored as a new memory. Let us consider eventualities 5-7 in more detail.

2.1 The newly input MR (or a component of it) is related to a stored memory via ontological subsumption, meronymy or location

There is much variability in the use of language, which can result from lack of knowledge of more precise terminology or from a person's understanding that certain kinds of underspecificity are entirely acceptable. For example, one can say *Let's eat at your **place*** rather than specifying whether we mean a house, a condominium or a studio apartment; and one can say *Does your arm hurt?* rather than asking *Does the broken bone in your arm hurt* or, even more specifically, *Does your ulna hurt?*

When attempting to match new textual input with a stored memory, the question is, how close do the compared MRs have to be in order to be considered a match? An important consideration when making this judgment is the application. In the dialog application we are developing, the notion of **sincerity conditions** plays an important role. That is, the VP expects the physician to ask it questions that it can answer; therefore, it should try hard — and search broadly, if necessary — to come up with the closest memories that will permit it to generate a response. In McShane et al. (2008) we suggest an algorithm that determines when two closely related MRs are close enough to be considered identical. The algorithm involves following three types of ontological links — subsumption, meronymy and location; if a match is found within the "lower" (i.e., domain-specific) ontology, then the related elements are considered a paraphrase. Let us show how this paraphrase processing works using a concrete example.

The physician asks the virtual patient, *Do you have any discomfort in your esophagus?* The MR for that question is as follows.

```
(REQUEST-INFO-1
   (THEME MODALITY-1.VALUE))
(MODALITY-1
   (TYPE EPISTEMIC)
   (SCOPE DISCOMFORT-1))
(DISCOMFORT-1
   (EXPERIENCER HUMAN-1)
   (LOCATION ESOPHAGUS-1))
(ESOPHAGUS-1
   (PART-OF-OBJECT HUMAN-1))
```

The interrogative mood gives rise to the instance of request-info, whose theme is the value of epistemic modality that scopes over the proposition headed by DISCOMFORT-1. (If the event actually happened, then the value of epistemic modality is 1; if it did not happen, then the value is 0).

The event DISCOMFORT is experienced by HUMAN-1, which will be linked to a specific human (the interlocutor) via reference resolution (reference resolution is carried out on every referring expression in OntoSem). The LOCATION of the DISCOMFORT is the ESOPHAGUS of that HUMAN.

If we extract the core meaning of this question, abstracting away from the interrogative elements, we have:

```
(DISCOMFORT-1
   (EXPERIENCER HUMAN-1)
   (LOCATION ESOPHAGUS-1))
(ESOPHAGUS-1
   (PART-OF-OBJECT HUMAN-1))
```

Let us assume that the patient has stored memories about its discomfort in a different way, as an undifferentiated symptom in its chest:

```
(SYMPTOM-1
   (EXPERIENCER HUMAN-1)
   (LOCATION CHEST-1))
(CHEST-1
   (PART-OF-OBJECT HUMAN-1))
```

Note that the patient stores memories of interoception directly, translating the output of its physiological agent into a memory that is in keeping with its own ontology. This translation is necessary because the agent's own ontology is a "lay" ontology — one lacking highly specified medical subtrees. The ontology available to the physiological agent, by contrast, is an "expert" ontology that is rich enough in medical knowledge to support disease simulation and treatment (see McShane et al., 2008).

The MR components in boldface are the ones that must be matched. DISCOMFORT is a child of SYMPTOM, forming a subsumption link of only one jump. ESOPHAGUS has a LOCATION of CHEST, and they are both PART-OF-OBJECT the human in question. Therefore, according to our matching algorithm — in conjunction with the fact that the VP assumes sincerity conditions in its conversations with the physician — the VP's memory of this event sufficiently matches the physician's question and the VP can respond affirmatively to the question: i.e., the VP has a memory of the symptom the physician is asking about.

In discussing the next two paraphrase-oriented phenomena we will shift to a different application area not because the medical domain lacks examples, but because understanding them would require too much background knowledge. The application area we will posit is an agent that is a personal companion, with the agent's job being to uphold its end of an open-ended conversation.

2.2 The newly input MR is related to a stored memory as the latter's precondition or effect

Consider the following dialog snippet between an elderly woman, Anne, and an intelligent agent that serves as her "conversational companion":

Anne: You know, my husband and I went to Rome for our honeymoon.
Agent: Is that so?
Anne: Yes. We ate such great artichokes in Trastevere!

As the agent participates in this conversation, it creates and stores memories of the meaning of Anne's utterances. In analyzing Anne's final utterance, the agent should create a link between Anne and her husband being in Trastevere, and Anne and her husband traveling to Rome.

OntoSem produces the following core meaning representation for the statement about traveling to Rome:

```
(TRAVEL-EVENT-1
   (AGENT SET-1)
   (DESTINATION ROME)
   (TIME (< find-anchor-time)))
(SET-1
   (MEMBERS HUMAN-1 HUMAN-2))
```

Some details are omitted in the above structure, as they are not relevant to our exposition here. Find-anchor-time is a meaning procedure that triggers a search in the metadata or in the text itself for the time of the event, which is before the time of speech. The above meaning representation will be stored by the agent in its memory.

The meaning representation (again, omitting some details) for the statement about eating in Trastevere will be:

```
(INGEST
   (AGENT SET-1) ; coreferential with SET-1 above
   (THEME ARTICHOKE-1)
   (LOCATION TRASTEVERE)
```

```
    (TIME (< find-anchor-time)))
  (MODALITY-1
    (TYPE EVALUATIVE)
    (SCOPE ARTICHOKE-1)
    (VALUE 1)
    (ATTRIBUTED-TO HUMAN-1))
```

At this point the agent must check whether the above MR should be linked in the agent's memory to the memory of the travel event processed earlier. In this case, a match is found — that is, the agent can establish that a precondition of the second event is among the effects of the first one. Specifically, the linking occurs because:

1. the agent's ontology contains the description of a complex event (a script) TRAVEL-EVENT, where it is listed that an *effect* of traveling to X is being in X;

2. the agent's ontology contains the knowledge that a *precondition* for an INGEST event taking place at LOCATION Y with AGENT X is that X is at LOCATION Y;

3. the agent's fact repository contains the knowledge that Trastevere is a neighborhood in Rome.

2.3 The newly input MR is related to a stored memory via ontological paraphrase

The third source of paraphrase we will discuss is what we call ontological paraphrase. This occurs when more than one metalanguage representation means the same thing. In an environment with only artificial agents, where communication can be carried out without resorting to natural language, such paraphrase should be excluded to the extent possible. However, in environments (like MVP) where meaning representations can be generated from natural language, this eventuality is more difficult to avoid. This is because a) basic meaning representations are produced on the basis of lexicon entries for words and phrases appearing in the sentence; and b) a word or phrase can be used in a particular sentence to render a narrower or broader meaning than it has in general. Now, in creating basic meaning representations, OntoSem uses concepts that are listed in the lexicon entries for the appropriate senses of the input words (in this paper we do not describe OntoSem's approach to word sense disambiguation and determination of semantic dependencies), and these concepts cannot reflect broadening or narrowing usages of the word. A good example of this phenomenon is the following: One can say (a) *go to London by plane* or (b) *fly to London*, and these inputs will generate different MRs:

(a) (MOTION-EVENT-7 (DESTINATION London) (INSTRUMENT AIRPLANE))
(b) (AERIAL-MOTION-EVENT-19 (DESTINATION London)).

This is because the semantics of the appropriate sense of *go* is explained using the concept MOTION-EVENT, while the semantics of the appropriate sense of *fly* uses AERIAL-MOTION-EVENT. In the former structure, the head event instance is more general than in the latter. In fact, the corresponding ontological concepts stand in a direct subsumption relation. If one chooses to use a concept that is higher in the ontological hierarchy, one may have to add further overt constraints to the meaning representation (like the one about the INSTRUMENT of the MOTION-EVENT above). If one chooses the lower-level, narrower ontological concept to start with, such constraints may be inherent in its definition (as is the case with AERIAL-MOTION-EVENT). This preference is the inverse of the lexical choice in text generation off of text meaning representations (for details see Nirenburg and Nirenburg, 1988).

OntoSem can yield either of the above basic text meaning representations. In many applications — for example, in interlingua-based machine translation — this would be quite benign. However, it is possible to create extended meaning representations such that the above variability is eliminated. The method we use for this purpose relies on the dynamic tightening or relaxation of selectional restrictions and is described in detail in Mahesh et al. (1997). Note that different paraphrases will still be produced for inputs that, while referring to the same event instance, describe it with a different degree of vagueness or underspecificity (see Section 2.1 above).

The fact that the two meaning representations above are paraphrases of one another can be automatically detected using a fairly simple heuristic: the ontological description of AERIAL-MOTION-EVENT includes the following property-value pairs:

```
AERIAL-MOTION-EVENT
  IS-A MOTION-EVENT
  INSTRUMENT AIRPLANE HELICOPTER BALLOON
```

Since the head of one of the MRs is an ancestor of the other, and the property-value pairs in the ancestor-based MR unify with the ontological definition of the descendant (the head of the other MR), these two structures are deemed to be paraphrases.

As we see from this example, world knowledge stored in the ontology is leveraged to carry out the reasoning needed to detect that the abovementioned formal structures are paraphrases. Such situations are somewhat similar to "bridging references" in the literature devoted to reference resolution (e.g. Poesio et al., 2004) because a knowledge bridge is needed to aid in the reference resolution of the entity.

A common source of this type of paraphrase derives from decisions about how to build the ontology. Ontology building is a complex task with "the lesser of the evils" decisions to be made at every turn. Two ontologies can be equally valid and yet look quite different. One of the most difficult aspects of ontology building is deciding when a new concept is needed. Let us continue with the example of taking a trip. A small excerpt from the MOTION-EVENT subtree of our ontology is as follows:

```
MOTION-EVENT
  AERIAL-MOTION
  LIQUID-CONTACT-MOTION
  SURFACE-CONTACT-MOTION
  TRAVEL-EVENT
  ...
```

As we can see, rather than having a single MOTION-EVENT lexically supplemented by property-value pairs that distinguish between types of motion, we have various types of motion being represented as different ontological concepts. This means that when different kinds of motion are referred to — even if they describe the same real-world event — they will instantiate different concepts in MR and we will be faced with the problem of matching at the level of MR. Whereas this matching problem can be seen as a vote for constraining the number of ontological concepts, there are practical reasons for not wanting to overdo this: for example, MRs are much harder to read and evaluate when lexical senses are described using property-value pairs rather than simply pointing to an iconic ontological concept that holds the description. Of course, the use of iconic concepts results in lower expressive power of an ontology, which affects the reasoning capabilities of agents in memory management, goal- and plan-based reasoning and the more complex cases of language understanding.

2.4 Theoretical Notes

This work derives from the theoretical assumption that in order for agents to show truly intelligent behavior their memory must be well managed. What is actually stored as a memory, however, is a complex question. For example, if someone were to describe a trip to New York and never referred to it as "trip to NY" but rather said that he "was in NY" (and the interlocutor knows that he doesn't live there), the interlocutor might still save the memory as

```
(TRAVEL-EVENT-1
   (AGENT HUMAN-1)
   (DESTINATION New York)).
```

So the "grain size" of memories is a compelling and complex problem. However, we must deflect a deep study of the question *What is memory*, agreeing with Minsky (2006) that lingering over definitions that might never be truly precise does not support practical progress.

Describing this work in broad terms risks conveying the impression that it is trivial, either conceptually or in terms of implementation. In fact, both of these facets of the work are quite complex, involving extensive theoretical and practical decision-making at every step — one of the reasons, perhaps, why it is not being broadly pursued. The standard counterargument that resource development for such an approach is too expensive does not hold up when one considers the cost of semantically annotating corpora and then building machine learning engines to exploit such corpora. Another criticism of knowledge-based approaches is that they are too narrow in coverage. Indeed, one cannot cover broad corpora at a deep level all at once; however, the insights gained in carrying out this kind of work, and its potential to significantly enhance the current state of the art in NLP, are really quite exciting. And, of course, there are no a priori preferences for starting with breadth and striving for depth over the opposite strategy of starting with depth and striving for breadth.

One final point must be mentioned. The OntoSem environment that forms the substrate for the work described here is used for applications that are much broader than MVP. Recent applications include question-answering and information extraction. Thus, the ongoing development of the ontology, lexicon, fact repository and semantic analyzer benefits a range of application areas.

3 State of Development

We have developed a complete simulated physiological agent that covers diseases of the esophagus and is capable of realistic physiological responses to even unexpected interventions. We have developed a tool for the fast creation of large libraries of physiological agents that feature different diseases, different genetic and behavioral predispositions and different realistic disease progressions. We have implemented a cognitive agent capable of interoception, perception through language, goal- and plan-based reasoning (within the domain of doctor-patient interaction), memory management, (simulated) physical action and (real) verbal action. With respect to interoception, we have developed a simulation of how the cognitive agent (the cognitive side of the "double" agent) perceives signals (symptoms) from its physiological agent counterpart. Even though a single knowledge representation substrate is used for modeling both agents, the interoception simulation process involves paraphrase.

We have developed a set of knowledge resources covering relevant knowledge about the world (the ontology), past events remembered by the agent (the fact repository) and knowledge about language (represented, largely, in the agent's lexicon).

The operational OntoSem semantic analyzer is used as the basic tool for creating meaning representations from language inputs. The latter already reflect results of the resolution of many kinds of paraphrase as a matter of course.

Content specification for the agent's dialog turns is addressed in the implemented goal- and plan-based reasoning module of the cognitive agent. Surface generation of agent dialog turns has, at this point, been implemented in a limited fashion, on the basis of prefabricated open textual patterns linked to types of text meaning representations that are output by the content specification module. In the next release of MVP we intend to incorporate a text realizer such as YAG (McRoy et al., 2003).

Upcoming evaluations will continue to help to debug the system and the underlying knowledge; and, more importantly for the topic of this paper, data will be collected for an evaluation of the dialog component of the system.

The quality of the system's treatment of paraphrase will be judged at two levels. First, the appropriateness of the agent's dialog responses will provide a practical, application-based measure of the quality of paraphrase processing, though blame assignment for any miscues will pose a complication. Second, we will be able to directly inspect the agent's memory for traces of the resolution of paraphrase against remembered concept instances and judge the appropriateness of these results against human decisions. To facilitate this, we will provide textual glosses to meaning representations comprising the agent's memory. The above regimen is the most economical way to evaluate our approach because an important prerequisite for evaluating the performance of paraphrase resolution algorithms in our environment is the creation of the fact repository (i.e., the agent's memory), against which the comparisons and linking occur. In the proposed testing regimen, this memory will be augmented and managed as a result of the operation of the system itself, so that there will be no need for creating it just for the purposes of evaluation.

References

Barzilay, R. and K. McKeown (2001). Extracting paraphrases from a parallel corpus. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-2001).*

Boonthum, C. (2004). iSTART: Paraphrase recognition. In *Proceedings of the Student Research Workshop at ACL 2004*, pp. 31–36.

Brun, C. and C. Hagège (2003). Normalization and paraphrasing using symbolic methods. In *Proceedings of the Second International Workshop on Paraphrasing (IWP 2003).*

Ibrahim, A., B. Katz, and J. Lin (2003). Extracting structural paraphrases from aligned monolingual corpora. In *Proceedings of the Second International Workshop on Paraphrasing (IWP 2003).*

Katz, B. and B. Levin (1988). Exploiting lexical regularities in designing natural language systems. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING-1988).*

Lapata, M. (2001). A corpus-based account of regular polysemy: The case of context-sensitive adjectives. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2001).*

Lin, D. (2001). Extracting collocations from text corpora. In *Proceedings of the First Workshop on Computational Terminology*.

Lin, D. and P. Pantel (2001). DIRT - discovery of inference rules from text. In *Proceedings of the ACM SIGKDD Conference Conference on Knowledge Discovery and Data Mining*.

Mahesh, K., S. Nirenburg, and S. Beale (1997). If you have it, flaunt it: Using full ontological knowledge for word sense disambiguation. In *Proceedings of TMI-97.*

McRoy, S. W., S. Channarukul, and S. S. Ali (2003). An augmented template-based approach to text realization. *Natural Language Engineering 9*(4), 381–420.

McShane, M., S. Nirenburg, and S. Beale (2008). Two kinds of paraphrase in modeling embodied cognitive agents. In *Proceedings of the Naturally-Inspired Artificial Intelligence AAAI Fall Symposium.*

McShane, M., S. Nirenburg, S. Beale, B. Jarrell, and G. Fantry (2007). 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).*

Minsky, M. L. (2006). *The Emotion Machine*. Simon & Schuster.

Nirenburg, S. and I. Nirenburg (1988). A framework for lexical selection in natural language generation. In *Proceedings of COLING-88.*

Nirenburg, S. and V. Raskin (2004). *Ontological Semantics*. MIT Press.

Pereira, F., N. Tishby, and L. Lee (1993). Distributional clustering of English words. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics (ACL-1991)*.

Poesio, M., R. Mehta, A. Maroudas, and J. Hitzeman (2004). Learning to resolve bridging references. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*.

Sowa, J. F. (1983). *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley.