

Natural Language Understanding (NLU, not NLP) in Cognitive Systems

Marjorie McShane

■ *Developing cognitive agents with human-level natural language understanding (NLU) capabilities requires modeling human cognition because natural, unedited utterances are anything but neat and complete; so understanding them requires the ability to clean up the input and fill in the lacunae. Indeed, language inputs regularly feature complex linguistic phenomena such as lexical and referential ambiguity, ellipsis, false starts, spurious repetitions, semantically vacuous fillers, nonliteral language, indirect speech acts, implicatures, and production errors. Moreover, cognitive agents must be nimble in the face of incomplete interpretations since even people do not perfectly understand every aspect of every utterance they hear. This means that once an agent has reached the best interpretation it can, it must determine how to proceed — be that acting upon the new information directly, remembering whatever it has understood and waiting to see what happens next, seeking out information to fill in the blanks, or asking its interlocutor for clarification. The reasoning needed to support NLU extends far beyond language itself, including, nonexhaustively, the agent's understanding of its own plans and goals; its dynamic modeling of its interlocutor's knowledge, plans, and goals, all guided by a theory of mind; its recognition of diverse aspects of human behavior, such as affect, cooperative behavior, and the effects of cognitive biases; and its integration of linguistic interpretations with its interpretations of other perceptive inputs, such as simulated vision and nonlinguistic audition. Considering all of these needs, it seems hardly possible that fundamental NLU will ever be achieved through the kinds of knowledge-lean text-string manipulation being pursued by the mainstream natural language processing (NLP) community. Instead, it requires a holistic approach to cognitive modeling of the type we are pursuing in a paradigm called OntoAgent.*

Mainstream natural language processing (NLP) of the past 25 years has concentrated on the manipulation of text strings within the empirical — so-called knowledge-lean — paradigm, in which sophisticated statistical techniques operate over large corpora, often relying on manual annotations to seed the learning process. The empirical paradigm has been quite successful in achieving a useful level of results for certain types of applications, such as knowledge extraction, question answering, and machine translation between pairs of languages for which large parallel corpora are available. It has also advanced our under-

standing of statistical “big data” methods themselves, quite apart from their application to NLP. However, the knowledge-lean orientation has become so thoroughly ensconced that its shortcomings — such as the inability to compute and record meaning (the basis of natural language *understanding*, or NLU) — seem to occupy a blind spot for the field at large. This is unfortunate because NLP and NLU are different beasts entirely, and both have their place in the forward march of science and technology ... or at least they should. Unfortunately, NLU has gotten squeezed almost to the point of extinction.

Kenneth Church (2011) presents a compelling analysis of this state of affairs, describing the pendulum swings between rationalism and empiricism starting with the inception of the field of computational linguistics in the 1950s. He attributes the full-on embracing of empiricism in the 1990s to a combination of pragmatic considerations and the availability of massive data sources.

The field had been banging its head on big hard challenges like AI-complete problems and long-distance dependencies. We advocated a pragmatic pivot toward simpler more solvable tasks like part of speech tagging. Data was becoming available like never before. What can we do with all this data? We argued that it is better to do something simple (than nothing at all). Let's go pick some low hanging fruit. Let's do what we can with short-distance dependencies. That won't solve the whole problem, but let's focus on what we can do as opposed to what we can't do. The glass is half full (as opposed to half empty) (Church 2011, p. 3).

In this essay, aptly titled “A Pendulum Swung Too Far,” Church calls for the need to reenter the debate between rationalism and empiricism not only for scientific reasons, but for practical ones as well:

Our generation has been fortunate to have plenty of low hanging fruit to pick (the facts that can be captured with short ngrams), but the next generation will be less fortunate since most of those facts will have been pretty well picked over before they retire, and therefore, it is likely that they will have to address facts that go beyond the simplest ngram approximations (Church 2011, p. 7).

My framing of the current article dovetails with Church's must-read essay. I will point out a number of unmotivated beliefs whose veracity crumbles as soon as one scratches the surface. How these beliefs attained quasi-axiomatic status among the NLP community is a fascinating question, answered in part by one of Church's observations: that recent and current generations of NLPers have received an insufficiently broad education in linguistics and the history of NLP and, therefore, lack the impetus to even scratch that surface.

Unmotivated Axiom #1:

The Knowledge Bottleneck

The early work on knowledge-based language processing systems, which was inspired by the original

goals of AI dating back to the 1950s, had dwindled to a trickle by the mid-1990s. The main culprit was disillusionment with how difficult the automation of language understanding turned out to be (see Nirenburg and McShane [2016a] for a historical perspective). The one-liner scapegoat was the knowledge bottleneck: the reality that language understanding requires machine-tractable lexical and ontological knowledge, along with reasoners that can exploit it — all of which are expensive to build. Although knowledge-lean approaches purported to circumvent this problem, those that involve supervised learning — and many do — simply shift the work of humans from building lexicons and ontologies to annotating corpora. When the resulting supervised learning systems hit a ceiling of results, developers point to the need for more or better annotations. Same problem, different veneer. Moreover, as Zaenen (2006) correctly points out, the success of supervised machine learning for syntax does not promise similar successes for linguistic phenomena that are less well understood, such as reference resolution or mapping linguistic structures to the state of the world. In short, it is not the case that knowledge-based methods suffer from knowledge needs whereas knowledge-lean methods do not: the higher-quality knowledge-lean systems do require knowledge in the form of annotations. Moreover, all knowledge-lean systems avoid phenomena and applications that would require unavailable knowledge support. What do all of those exclusions represent? All of the issues whose solution is necessary to attain the next level of quality of automatic language processing.

Unmotivated Axiom #2:

Knowledge-Based Methods Were Tried and Failed

Yorick Wilks (2000) says it plainly: “... The claims of AI/NLP to offer high quality at NLP tasks have never been really tested. They have certainly not failed, just got left behind in the rush towards what could be easily tested!” Everything about computing has changed since the peak of knowledge-based work in the mid-1980s — speed, storage, programming languages, their supporting libraries, interface technologies, corpora, and more. So comparing statistical NLP systems of the 2010s with knowledge-based NLP systems of the 1980s says nothing about the respective utility of these R&D paradigms. As a side note, one can't help but wonder where knowledge-based NLU would stand now if all, or even a fraction, of the resources devoted to statistical NLP over the past 25 years had remained with the goal of automating language understanding.

Unmotivated Axiom #3:

NLU Is an Extension of NLP

Fundamental NLU has little to nothing in common with current mainstream NLP; in fact, I think it has

much more in common with robotics. Like robotics, NLU is most naturally pursued in service of specific tasks in a specific domain for which the agent is supplied with the requisite knowledge and reasoning capabilities. However, whereas domain-specific robotics successes are praised (and rightly so!), domain-specific NLU successes are criticized for not being immediately applicable to all domains — under the pressure of evaluation metrics entrenched in statistical NLP.¹ One step toward resolving this miscasting of NLU might be the simple practice of reserving the term *NLU* for actual deep understanding, rather than watering it down by applying it to any system that incorporates even cursory semantic or pragmatic features. Of course, marrying robotics with NLU is a natural fit.

Unmotivated Axiom #4:
It's Either NLP or NLU

One key to the success of NLP has been finding applications and system setups that circumvent the need for language understanding. For example, consider a question-answering system that has access to a large and highly redundant corpus. When asked to indicate when the city of Detroit was founded, it can happily ignore formulations of the answer that would require sophisticated linguistic analysis or reasoning (*It was founded 2 years later; That happened soon afterward*) and, instead, fulfill its task with string-level matching against the following sentence from Wikipedia: “Detroit was founded on July 24, 1701 by the French explorer and adventurer Antoine de la Mothe Cadillac and a party of settlers.”² However, not all language-oriented applications offer such remarkable simplifications. For example, agents in dialogue systems receive one and only one formulation of each utterance. Moreover, they must also deal with performance errors such as unfinished thoughts, fragmentary utterances, self-interruptions, repetitions, and non sequiturs. Even the speech signal itself can be corrupted, as by background noise and dropped signals. Consider, in this regard, a short excerpt from the *Santa Barbara Corpus of Spoken American English*, in which the speaker is a student of equine science talking about blacksmithing:

we did a lot of stuff with the — like we had the, um, ... the burners? you know, and you'd put the — you'd have — you started out with the straight iron? .. you know? and you'd stick it into the, into the, you know like, actual blacksmithing (Du Bois et al. 2000–2005).³

Unsupported by the visual context or the intonation and pauses of spoken language, this excerpt requires quite a bit of effort even for people to understand. Presumably, we get the gist thanks to our ontological knowledge of the context (I told you that the topic was blacksmithing). Moreover, we make decisions about how much understanding is actually needed before we stop trying to understand further. In sum, NLP has one set of strengths, purviews, and

methods and NLU has another. These programs of work are complementary, not in competition.

Unmotivated Axiom #5:
Whereas Mainstream NLP Is Realistic, Deep NLU Is Unrealistic

The miscomprehension here derives from an undue emphasis on compartmentalization. If one plucks NLU out of overall agent cognition, dangles it by itself, and expects meaning analysis to be carried out to perfection in isolation, then I would agree that the task is unrealistic. However, this framing of the problem is misleading.⁴ To understand language inputs, a cognitive agent must know what kinds of information to rely upon during language analysis and why; it must also use stored knowledge to judge how deeply to analyze inputs. Analysis can involve multiple passes over inputs, requiring increasing amounts of resources, with the agent pursuing the latter stages only if it deems the information worthy of the effort. For example, a virtual medical assistant tasked with assisting a doctor in a clinical setting can ignore incidental conversations about pop culture and office gossip, which it might detect using a resource-light comparison between the input and its active plans and goals. By contrast, that same agent needs to understand both the full meaning and the implications in the following doctor-patient exchange involving a patient presenting with gastrointestinal distress:

Doctor: “Have you been traveling lately?”

Patient: “Yes, I vacationed in Mexico two weeks ago.”

For further discussion of the need for integrated cognitive language processing — along with proven successes in a robotic implementation — see Lindes and Laird (2016).

Recap

To recap, I have just suggested that five misconceptions have contributed to a state of affairs in which statistical NLP and knowledge-based NLU have been falsely pitted against each other. But this zero-sum-game thinking is too crude for a domain as complex as natural language processing/understanding. The NLP and NLU programs of work pursue different goals and promise to contribute in different ways, on different timelines, to technologies that will enhance the human experience. Clearly there is room, and a need, for both.

In the OntoAgent paradigm in which I work, we are as tantalized by the prospects of human-level language understanding as were the founders of AI over a half century ago. We find it fascinating that language strings represent only the tip of the iceberg of language meaning, and that every speaker of every language is a master of effortlessly recreating a functionally sufficient approximation of the whole iceberg. Endowing agents with such immense capabilities is challenging. This article identifies some of the

linguistic challenges faced by cognitive agents tasked with fully interpreting natural language, and it explains the knowledge-based, reasoning-oriented modeling strategies used to address them in the OntoAgent cognitive architecture. In this article I will focus on select examples, keep formalisms to a minimum, and omit details that can be found in cited publications. I will show that modeling cognition requires an integrated approach to NLU, one that allows us to use many types of language analysis to attain a high quality of results that would be difficult or even arguably impossible to achieve with string-level processing or narrow task- or method-driven approaches. Finally, I will suggest that fundamental NLU should return to the central agenda of AI, not to compete with mainstream NLP on what the latter does well, but to give rise to a new generation of more sophisticated language-endowed intelligent agents (LEIAs).

A question that will naturally arise with respect to the descriptions that follow is, “Has all of this been implemented?” The short answer, which I will expand upon at the end, is that much of it has been implemented, with various levels of coverage of sub-phenomena. But more importantly, we have demonstrated, through a combination of implementation and formal algorithm specification, that none of what we do requires a “magic happens here” step: we have developed methods to operationalize all of these capabilities and are testing these methods in implementations that use nontoy knowledge bases. No doubt, additional knowledge engineering (the expansion of the lexicon, ontology, rule sets) is always useful and will be an ongoing concern; but knowledge engineering is just work, involving relatively straightforward variations once the conceptual and methodological themes have been established. Our hope that this work will garner the enthusiasm, support, and resources to allow it to flourish is fueled by our knowledge that eminent thinkers such as Annie Zaenen (2006), Ray Jackendoff (2007), Kenneth Church (2011), and John Laird (Laird, Lebiere, and Rosenbloom 2017), to name just a few, have been voicing compatible views about the need for deep language understanding for language-endowed cognitive agents.

Ontological Semantics for LEIAs

NLU in OntoAgent follows the theory of Ontological Semantics (OS; Nirenburg and Raskin 2004). The goal of OS language understanding is to generate contextually disambiguated, ontologically grounded text meaning representations that are stored to agent memory in support of subsequent reasoning about action. Individual linguistic phenomena are treated by microtheories which, at any given time, can be at various stages of advancement both descriptively and in terms of implementation. For example, recently

published OS microtheories address topics such as lexical disambiguation (McShane, Nirenburg, and Beale 2016), multiword expressions (McShane, Nirenburg, and Beale 2015), nominal compounds (McShane, Beale, and Babkin 2014), verb phrase ellipsis (McShane and Babkin 2016a), and unexpected input (McShane, Blissett, and Nirenburg 2017).

The OS language analyzer is supported by a 30,000-sense semantic lexicon and a 9,000-concept, property-rich ontology. Although knowledge bases of this size do not provide comprehensive coverage of all domains, they are a far cry from the so-called toy systems of early AI, and they serve as a realistic test case for our theories and methods. For example, the OS lexicon has several dozen senses of the word *take*, most of them idiomatic or construction-like (*take pity on*, *take a shower*, *take leave of*, and so on). Each sense is supplied with lexical, syntactic, and semantic constraints, which, in the best case, allow the analyzer to converge on exactly one disambiguation decision per context. However, given that most sentences contain more than one multiply ambiguous word, and given that extra-sentential context is often needed for disambiguation, other heuristic evidence can be required to arrive at an interpretation, such as reference resolution, contextually triggered ontological scripts, the plans and goals on the agent’s agenda, and even the results of processing inputs through other channels of perception, such as simulated vision.

Although OS language analysis methods can be applied to any domain, the approach is better suited to applications for which additional knowledge of the types just mentioned can be recorded. This reflects the fact that language understanding, for humans and machines alike, is never just about the words in a sentence, it always requires background knowledge and contextual awareness. A recent prototype system that demonstrated how OS language understanding can contribute to overall agent functioning within a defined domain is the Maryland Virtual Patient prototype physician training system, in which a cohort of intelligent agents — endowed with both physiological and cognitive simulations — served as virtual patients who could be diagnosed and treated by system users (Nirenburg, McShane, and Beale 2008; McShane and Nirenburg 2012; McShane et al. 2012; McShane, Nirenburg, and Jarrell 2013).

Reasoning During Language Understanding

This section presents a sampling of the types of reasoning applied to language understanding by language-endowed intelligent agents (LEIAs) in OntoAgent, starting with the simplest example and progressing to more complicated cases.⁵ In all instances, the reasoning is inspired by our hypotheses about human cognition, but the implementations reflect computationally expedient ways of realizing

humanlike results. Readers who choose to skim through the technical details are nevertheless encouraged to reflect on how much reasoning people actually apply to language understanding — automatically, without effort, and usually without even noticing that there is anything to reason about.

Matching Recorded Constraints

The simplest case of language analysis is illustrated by the example *A brown squirrel is eating a nut*. For this input, like any other, the LEIA must disambiguate each lexeme (that is, understand it as an instance of a particular concept in its ontology) and combine those interpretations into an overall semantic representation like the one in figure 1.

The representation in figure 1 is read as follows. The first frame is headed by a numbered instance of the concept `INGEST`, concepts being distinguished from words of English by the use of small caps.⁶ `INGEST-1` has three contextually relevant property values: its `AGENT` (the eater) is an instance of `SQUIRREL`, its `THEME` (what is eaten) is an instance of `NUT-FOODSTUFF`, and the `TIME` of the event is the time of speech, which must be computed by the agent, if possible, using the procedural semantic routine *find-anchor-time* (this routine has not yet been launched at the stage of analysis shown here). The next frame, headed by `SQUIRREL-1`, shows not only the inverse relation to `INGEST-1`, but also that the `COLOR` of this `SQUIRREL` is `BROWN`. Since we have no additional information about the nut, its frame — `NUT-FOODSTUFF-1` — shows only the inverse relation with `INGEST-1`. Developer views of text meaning representations also include many types of metadata, such as which word of input gave rise to each frame, which lexical sense provided the given interpretation, and so on.

Abstracting away from details of particular implementations of OS,⁷ let us work through the analysis process. First the input is syntactically parsed using a parser developed externally from our system.⁸ Then the LEIA attempts to align the parse with the syntactic expectations recorded in the lexicon for the words in the sentence. For example, when it looks up the verb *eat*, it finds three senses: one is optionally transitive and means `INGEST`; the other two describe the idiom *eat away at* in its physical and abstract senses (*The rust ate away at the pipe; His behavior is eating away at my nerves!*). Since the idiomatic senses require the words *away at*, which are not present in our input, they are rejected, leaving only the `INGEST` sense as a viable candidate.⁹ A simplified version of the needed lexical sense is shown in figure 2.

The syntactic structure (`syn-struc`) zone says that this sense of *eat* is optionally transitive: it requires a subject and can be used with or without a direct object. The semantic structure (`sem-struc`) zone says that this sense of *eat* means `INGEST`. Each constituent of input is associated with a variable in the `syn-struc`: the subject is `$var1` and the direct object is `$var2`.

INGEST-1	
AGENT	SQUIRREL-1
THEME	NUT-FOODSTUFF-1
TIME	<i>find-anchor-time</i>
SQUIRREL-1	
COLOR	BROWN
AGENT-OF	INGEST-1
NUT-FOODSTUFF-1	
THEME-OF	INGEST-1

Figure 1. The Text Meaning Representation for A Brown Squirrel Is Eating a Nut.

eat-v1: "ingest," as in "He was eating (cheese)."			
syn-struc		sem-struc	
subject	\$var1	INGEST	
root	\$var0	AGENT	^\$var1
directobject	\$var2 (optional)	THEME	^\$var2

Figure 2. A Simplified Version of the `INGEST` Sense of *Eat*.

Those variables are linked to their semantic interpretations in the `sem-struc` (^ is read as "the meaning of"). So the word that fills the subject slot in the `syn-struc` (`$var1`) must first be semantically analyzed, resulting in `^$var1` (the meaning of `$var1`); that concept can then be used to fill the `AGENT` role of `INGEST`. For example, given our input *A brown squirrel is eating a nut*, the LEIA links the word *squirrel* to `$var1` then semantically analyzes it as `SQUIRREL` before using it to fill the `AGENT` role of `INGEST`. An analogous process occurs for `$var2/^$var2`. The ontology, for its part, constrains the valid fillers of the case-roles of `INGEST` as shown in figure 3.

This excerpt from the ontological frame for `INGEST` — which actually contains many more properties and expectations about their values — says that its typical `AGENT` (that is, the basic semantic constraint indicated by the *sem* facet) is an `ANIMAL`; however, this constraint can be relaxed to `SOCIAL-OBJECTS` (for example, *The fire department eats a lot of pizza*). Similarly,

INGEST		
AGENT	sem	ANIMAL
	relaxable-to	SOCIAL-OBJECT
THEME	sem	FOOD, BEVERAGE, INGESTIBLE-MEDICATION
	relaxable-to	ANIMAL, PLANT
	not	HUMAN

Figure 3. An Excerpt from the Ontological Frame for *INGEST*.

the description of the *THEME* indicates that *FOOD*, *BEVERAGE*, and *INGESTIBLE-MEDICATION* are the most typical *THEMES*, but other *ANIMALS* and *PLANTS* not already subsumed by the *FOOD* subtree might be consumed as well. *HUMANS* are explicitly excluded as ingestibles using the *not* facet since they would otherwise be understood as unusual-but-possible ingestibles due to their placement in the *ANIMAL* subtree.¹⁰

Having narrowed down the interpretation of *eat* to a single sense, the LEIA must now determine which senses of *squirrel*, *brown*, and *nut* best fit this input. *Squirrel* and *brown* are easy: the lexicon contains only one sense of each, and these senses fit well semantically (*SQUIRREL* is a suitable *AGENT* of *INGEST*, and *BROWN* is a valid *COLOR* of *SQUIRREL*). However, there are three senses of *nut*: an edible foodstuff, a crazy person, and a machine part. We just saw that neither people nor machine parts are suitable *THEMES* of *INGEST*, leaving only the *NUT-FOODSTUFF* sense, which perfectly fits the ontological constraints of *INGEST* and is selected as a high-confidence interpretation.

Operationally speaking, the meaning representation for *A brown squirrel is eating a nut* is generated by (1) copying the *sem-struct* of *eat-v1* into the nascent text meaning representation; (2) translating the concept type (*INGEST*) into an instance (*INGEST-1*); and (3) replacing the variables with their appropriate interpretations (*SQUIRREL-1* [*COLOR BROWN*], *NUT-FOODSTUFF-1*). In terms of run-time reasoning, this example is as simple as it gets since it involves only constraint matching, and all constraints match in a unique and satisfactory way. “Simple constraint matching” does not, however, come for free: its precondition is the availability of high-quality lexical and ontological knowledge bases that are sufficient to allow the LEIA to disambiguate and validate the semantic congruity of its interpretations.

Incorporating Contextual Clues

The combination of lexical, syntactic, and semantic constraints within the local dependency structure does not always lead to a single high-confidence interpretation of an input. For example, *He kicked the*

bucket could refer to dying or making foot contact with a container. Both of these interpretations are provided for by the OS lexicon: there is an idiomatic sense of *kick* that requires the direct object to be *the bucket* and means *DIE*; and there is a physical-action sense that can take any physical object as its *THEME* and means *KICK*. Disambiguation requires context, but “using the context” is impotently vague until grounded in machine-tractable heuristics. Luckily, in this case, we have a point of traction, but understanding it requires appreciating the multifunctionality of the easily overlooked word *the* in English.

Apart from indicating coreference with an already-introduced object (*A dog loped into the room, startling the resident cat. Luckily, the dog turned out to be friendly*), *the* is used in proper names (*the CIA*), in idioms (*on the one hand ... on the other hand; kick the bucket*), in so-called “universally known” objects (*the sun, the solar system*), in superlatives (*the best ice cream in the world*), in ordinals (*the second barn on the left*), and more. So, every time a LEIA encounters the word *the* it must ask, “Does this indicate coreference, or is its use licensed in some other way?” To decide, the LEIA searches the context for an available coreferent. If it finds a strong candidate, it creates the coreference link and selects the associated interpretation; if not, it uses an interpretation that does not require an antecedent. Let us trace how this works using the examples that follow.

- (1) Be sure your paint is in a tall bucket and on a tarp so that if you accidentally kick the bucket you won't stain the floor.
- (2) Claude tried to milk the cows quietly so as not to wake everyone up, but he kicked the bucket, which got the dogs barking, which roused the whole house.

When processing (1), the LEIA will evaluate and score all candidate antecedents for “the bucket”: *your paint, a tall bucket, a tarp*. The candidate “a tall bucket” (analyzed as [*BUCKET: HEIGHT .8*]) will receive a very high coreference score since it is nearby and maps to the same ontological concept. This coreference score will be combined with the high semantic score for the literal interpretation of *kick the bucket* since the physical interpretation of *BUCKET* is a perfect *THEME* of *KICK*. The resulting high score for the contextually sensitive, literal interpretation of *kick the bucket* will win out over the idiomatic reading, since the latter will be penalized for having to ignore the available coreferent for *the bucket*.

Example (2) also introduces a bucket into the context, but it does so implicitly. Even nonfarmers know that to milk a cow one takes a bucket, puts it under the cow's teats and squeezes them to release the milk into the bucket. Therefore, as soon as cow milking enters the context, a bucket is virtually available. How can we be sure that *bucket* has been virtually introduced into the context? By the fact that we can refer to it using *the!* Consider some analogous contexts: *On my flight to Hawaii the pilot talked a lot; Oh,*

1. Definition: A professional advises someone.
 Example: My doctor sees patients every day from 8 till 5.
 Constraints: The subject indicates a PROFESSIONAL and the direct object indicates a HUMAN.
 Meaning: PROVIDE-PROFESSIONAL-CONSULTATION
 AGENT PROFESSIONAL
 BENEFICIARY HUMAN

2. Definition: Someone consults with a professional.
 Example: I see my doctor regularly.
 Constraints: The subject indicates a HUMAN and the direct object indicates a PROFESSIONAL.
 Meaning: PROVIDE-PROFESSIONAL-CONSULTATION
 AGENT PROFESSIONAL
 BENEFICIARY HUMAN

3. Definition: Someone visually perceives something.
 Example: I see a blue car.
 Constraints: The subject indicates an ANIMAL and the direct object indicates a PHYSICAL-OBJECT.
 Meaning: INVOLUNTARY-VISUAL-EVENT
 AGENT ANIMAL
 THEME PHYSICAL-OBJECT

Figure 4. Three Senses of See in the OS Lexicon.

dreaded lectures when the speaker mumbles!; *If you buy an old car, don't expect the air conditioning to work.* LEIAs can carry out this same kind of reasoning if their ontologies are supplied with scripts — that is, typical sequences of events and their participants (Schank and Abelson 1977). In order to find a script-based justification for a *the*-phrase, the agent must search its ontology for all of the events mentioned in the immediate context and see if the given object is listed among the participants. Finding one both justifies the use of *the* and suggests a literal interpretation of the object in the context. If, by contrast, the agent does not find a strong coreferent for *bucket*, it will happily use the idiomatic reading (DIE) that does not require one.

Leveraging Rules of Thumb

When recorded constraints, reference resolution, and ontological scripts are still not sufficient to disambiguate an input, rules of thumb can sometimes help — as long as they are appropriately understood as

defeasible preferences. Consider the sentence *My grandmother saw her doctor yesterday*. Outside of context, the default interpretation is that my grandmother attended a medical consultation provided by her doctor. However, my grandmother could also have simply caught sight of her doctor, an interpretation preferred if we add the modifier *at the beach*. Moreover, if we really beef up the context, we can turn grandma into the person providing the professional consultation: *My grandmother is the leading lawyer in the city. Everybody looks to her for advice. In fact, she saw her doctor yesterday and Dad's chiropractor day before, both of whom were dealing with malpractice suits.*

Consider how a LEIA can use rules of thumb to help its understanding of these contexts. Among the many senses of *see* in the OS lexicon, three are relevant here. Informally, they are depicted in figure 4 (note that *I saw my doctor* and *My doctor saw me* can mean the same thing in the professional-consultation sense):

Without further contextual clues or background knowledge, the input *My grandmother saw her doctor yesterday* will be accommodated by senses 2 and 3; it will not match sense 1 because there is no evidence in this minimal context that grandma meets the PROFESSIONAL constraint on the subject. The LEIA's rule of thumb for selecting a default interpretation among these is to prefer the interpretation with the tighter semantic constraints on its arguments. In this case, sense 2 wins since its subject/object constraints are HUMAN/PROFESSIONAL versus sense 3's broader constraints ANIMAL/PHYSICAL-OBJECT. This rule of thumb is inspired by, and aligns with, human intuitions about the default interpretation of this sentence.

However, this rule of thumb only applies if all properties of the context align with basic ontological expectations. This is not the case in *My grandmother saw her doctor at the beach* because *the beach* is not a typical location for a professional consultation — information that is available in the ontology. Based on this, the agent must reject the narrower interpretation and opt for the broader one, INVOLUNTARY-VISUAL-EVENT, which violates no ontological expectations. As for the context in which lawyer-grandma consults her doctor — that requires a type of dynamic analysis of saliency that is on agenda to be modeled.

Learning New Words and Concepts

Language understanding often results in learning. A straightforward type of learning involves learning new facts about already-known types of objects and events. For example, if a LEIA knows that people can get the flu, and it knows various ways of expressing “getting the flu” in English, then it can readily learn the fact that John got the flu yesterday. More complicated is learning about new kinds of objects and events along with the words and phrases used to describe them. In the Maryland Virtual Patient (MVP) clinician training application mentioned earlier, LEIAs playing the role of virtual patients carried out such learning in dialogues with human users.

Consider figure 5, which depicts an example extracted from the middle of a system run at a point when the user, playing the role of clinician, has enough information about the virtual patient to suspect an esophageal disease. He recommends that the patient agree to have a diagnostic procedure called an EGD. In the excerpt, the utterances themselves are presented in semibold font. The italics indicate traces of system functioning, presented in human-readable form for demonstration purposes (the agent “thinks” in the ontological metalanguage, certainly not in English!).

When the virtual patient receives each of the doctor's dialogue turns as input, it analyzes it, generating an ontologically-grounded text meaning representation of the type described earlier. This not only supports the learning of ontology and lexicon, it also feeds into the agent's decision function about

whether or not to agree to the procedure. This particular virtual patient has character traits that block it from agreeing to any advice without having sufficient information to make an informed decision (for more on our modeling of character traits, see McShane 2014). This virtual patient is particularly concerned about the risk and pain of the recommended procedure and asks about them. It knows that procedures can have RISK and PAIN because all MEDICAL-PROCEDURES in its ontology inherit these properties. Only when satisfied with the levels of risk and pain does the patient agree to the procedure. In short, the LEIA's learning of lexicon and ontology — which is an essential part of its language understanding capabilities — is both driven by and contributes to its reasoning about action.

Resolving Ellipsis and Interpreting Fragments

Natural language is highly elliptical and would be gruelingly longwinded otherwise. Some types of ellipsis, such the ellipsis of verb phrases (example 3) are found in all language genres, whereas other types, such as fragmentary rejoinders (example 4), are particularly common in dialogues.

(3) Aid workers in war-ravaged Kabul were stunned when a toddler from a poor family offered to *teach illiterate women to read and write* — and then promptly proved he could [e]. (Graff and Cieri 2003)¹¹

(4) “Everybody has to help clean up.” “Not me.”

No matter the type of ellipsis, both stages of its processing — detecting the gap and reconstructing its meaning — can be quite difficult. One of our methods for treating ellipsis in the near- to midterm is teaching LEIAs to independently detect which instances they think they can treat with high confidence and focus their resources on those.¹² As for the residual cases, agents can react using their usual repertoire of moves: ask a human for clarification, wait and see if upcoming utterances make things clear, and so on. Although there are many types of elliptical and fragmentary utterances, three will be sufficient for illustration: verb phrase ellipsis, fragments in question-answer pairs, and stand-alone fragments.

Verb phrase ellipsis is the nonexpression of a verb phrase whose meaning can be reconstructed from the context. For example, the elided verb phrase in example 5 means *be in the kitchen*.

(5) “We’re celebrating the fact that we’re living in a time where, when we want to be in the kitchen, we can [e],” says Tamara Cohen, Ma’yan program director. (Graff and Cieri 2003)

Our verb phrase ellipsis module (McShane and Babkin 2016a) uses lexical and syntactic heuristics to select which contexts it knows how to treat, and it identifies the antecedent for those cases. Compare example 5, which the system can treat, with example 6, which it cannot.

(6) The former Massachusetts governor called on Unit-

MD: **I suggest having an EGD, which is a diagnostic procedure.**

VP: *I do not know the word "EGD".
I record the concept EGD as a child of DIAGNOSTIC-PROCEDURE in my ontology.
I record the word "EGD" as a noun in my lexicon, mapped to the concept EGD.
I don't know enough about the properties of EGD to agree.
I need to ask questions about RISK and PAIN.
How risky is it?*

MD: **It's not risky at all.**

VP: *I add the property-value pair "RISK: 0" to the ontological specification of EGD.
Is it painful?*

MD: **It's only a little uncomfortable.**

VP: *I add the property-value pair "PAIN: .1" to the ontological specification of EGD.
I can tolerate that amount of risk and pain.
OK, I'll agree to that.*

Figure 5. Example Maryland Virtual Patient Clinician Training Application System Run.

ed Nations Secretary General Ban Ki-moon to *revoke Ahmadinejad's invitation to the assembly* and warned Washington should reconsider support for the world body if he did not [e] (Graff and Cieri 2003).

Both contexts are complex enough to offer multiple candidate antecedents for the ellipsis: in example 5 the elided verb phrase could be headed by *celebrate, live, want or be*; and in example 6 it could be headed by *call on, revoke, warn or reconsider*. However, whereas example 5 can be successfully simplified using automatic syntactic tree trimming procedures, example 6 cannot. Specifically, when processing example 5, the system can leverage a pruning function that crosses out the material prior to the first comma, leaving a much simpler context from which to select the antecedent. Since such generic text simplification procedures are not applicable to example 6, the context remains complex and ambiguous. Once the string-level antecedent in example 5 has been identified, its semantic analysis is incorporated into the overall text meaning representation. This involves not only concept selection but also the determination of whether there is a type-coreference or instance-coreference relationship between the antecedent and the elided category.

Another type of ellipsis is characterized by fragments occurring in typical dialogue strategies, such as

question-answer pairs. As described in McShane, Nirenburg, and Beale (2005), the text meaning representation of a question (*How much ice cream do you eat every week?*) includes the expectation that its answer will follow (*A half gallon*). So, when the question-answer pair occurs in sequence — which, however, is far from always the case in real language use — incorporating the meaning of the fragment into the meaning of the overall context is straightforward.

Although the aforementioned strategies are largely domain-independent, they are best supplemented by domain-sensitive ones, when available. For example, imagine that a surgeon, assisted by a LEIA robot, yells to the robot, "Scalpel!" We know that the surgeon wants to be handed a scalpel, but how do we prepare the robot to understand that? On the one hand, we could write a rule saying that "Scalpel!" always means "Hand me a scalpel"; but unless the robot has an extremely narrow repertoire of capabilities, this type of listing will be inefficient and ultimately unsatisfactory. Zooming out one level of abstraction, we could generalize that "PHYSICAL-OBJECT!" always means "Hand me a PHYSICAL-OBJECT." This will often work — except when it doesn't: *Nuts! Lawyers! My foot!* Constraining the applicable objects still further to IMPLEMENTS would help — and, in fact, it might be entirely sufficient for our robot. However, a more fundamen-

tal solution would be for the agent to figure out what was needed through reasoning about its collaborator's plans and goals, in conjunction with its understanding of its own capabilities. If all the robot can do is hand objects to humans, then there is no language analysis problem: as long as the robot can identify, reach, and lift the object, it knows what to do. But if, by contrast, the robot can do many things with the same set of objects — hand them to someone, hold them steady in the given position, turn them to the right, replace them to their storage area — contextual cues have to be incorporated into the language understanding process. There is nothing particularly difficult about specifying these cues, but it cannot be done generically, it must be done for each robot (or unembodied intelligent agent) functioning in each context of interest. To reiterate a point made earlier, in order to function usefully in an application, intelligent agents need knowledge about their domain, their own capabilities, their collaborators' needs, and so on, and this knowledge can efficiently be reused to support language understanding.

Interpreting Utterances Incrementally

NLP has most often been approached by subjecting full sentences to a pipeline of processing: preprocessing followed by syntactic analysis and whichever aspects of semantic and pragmatic analysis might be undertaken (often none). By contrast, we have recently begun exploring incremental NLU, defined as building up meaning representations using whatever heuristic evidence is available as early as possible. This not only more closely emulates what people do,¹³ it should allow for time-sensitive actions by the agent, such as interrupting for clarification and starting to take action before the speaker has finished a long sentence: By the time one says “Grab the fire extinguisher” the robot should already be on its way, no matter what subsequent instructions might follow.

Our approach to incrementality is inspired by what we think people do but with a large dose of system-building practicality. For example, although syntactic analysis is run on each new word of input, semantic analysis is run only if the last word is a noun or a verb — that is, a semantically heavy element. So the LEIA will attempt semantic analysis at each of the points indicated by an asterisk in the following input: *The green machine * makes * great coffee * in no time.** This strategy operationalizes our opinion that there is no added value in forcing the agent to attempt to semantically analyze *The* or *The green* in isolation, no matter what people may or may not do with such fragments.

As an example of incremental analysis, consider the following inputs:

- (7a) The truck delivered the soil and then it
- (7b) The truck delivered the soil and then it took
- (7c) The truck delivered the soil and then it took the lumber away.

Even given only the string in input 7a, the LEIA can

hypothesize that *it* most likely corefers with *the truck*. The main heuristics it uses to reach this judgment are *the truck* and *it* have matching features (inanimate, singular); they are the subjects of sequential clauses; those clauses are joined by the conjunction-adverb pair *and then*. Our experimentation has shown that this configuration highly suggests that *the truck* and *it* are coreferential (for details of this and other predicative coreference configurations, see McShane and Babkin [2016b]). Moving to example 7b, once the agent has hypothesized that *it* refers to *the truck*, it can penalize all senses of *take* whose subject must be either pleonastic (that is, nonreferential, as in *It takes time to learn things well*) or refer to a HUMAN. This aids in word-sense disambiguation because *take* is a light verb that has dozens of senses. Of course, subsequent words of input can always alter the agent's current analysis: for example, 7a could have continued as “... and then it started to rain.” In this case, the agent would have to overturn its early coreference link between *it* and *the truck* based on the new evidence.

A major reasoning challenge in this approach involves weighing competing preferences posted by different analysis modules. Consider in this regard example 8:

- (8) The boy's father talked at length with the surgeon and then he proceeded to operate.

Readers might find this sentence rather awkward since, like our LEIAs, they might intuitively expect “and then he” to signal coreference with the preceding clause's subject and then have to switch their interpretation midstream. In fact, an editor would likely improve this sentence by avoiding the suboptimal “he” in favor of something less ambiguous, like “the latter.” However, faced with exactly this sentence — since it could realistically be pronounced by a person in regular discourse — the LEIA must override its syntactically grounded preference for the *father/he* coreference by prioritizing the semantic preference for the AGENT OF SURGERY to be a SURGEON.

Managing competing heuristic evidence is challenging. The LEIA must not only estimate confidence in each of its individual calculations (each word sense, each coreference decision), it must weigh that evidence to arrive at a single overall analysis. Our agent's advantage in cutting through the complexity is that its language understanding takes place within a task-oriented context that provides it with expectations about what the interlocutor is likely to say, want, and need. Those expectations can guide decision-making in what might otherwise be an infeasible amount of residual ambiguity.

Reasoning About the Results of Language Understanding

We have just considered a handful of language understanding challenges and how LEIAs address them. In passing, we have mentioned some key decision

points involving reasoning about the results of language understanding. Now let us consider the latter more fundamentally.

Learning Words and Concepts

Every word or phrase an agent encounters in an input can have various statuses: it can be a known word/phrase used in a known sense; a known word/phrase used in a canonical sense that simply hasn't been recorded in the lexicon yet; a known word/phrase used in a noncanonical sense — for example, as a metonymy; or an unknown word/phrase. In all but the first case, the agent faces incongruity resulting in an incomplete meaning representation, and it must decide what to do next. Let us concentrate on the case of completely unknown words, with the understanding that unknown senses of known words — and all corresponding eventualities for phrases — can be handled similarly.

If learning words of a particular ontological type is among the agent's active goals, then it will go ahead and learn the new word — as explained earlier using the example of EGD. If, by contrast, the given type of word does not have special priority in the application, then the agent can attempt some straightforward inferencing techniques to glean at least a coarse-grained meaning. For example, given an input like *The jiffers took a bath*, the agent can infer that *jiffers* is probably some sort of HUMAN or, at a minimum, some ANIMAL, based on the combination of lexical knowledge (*take a bath* is a phrase that points to the concept BATHE) and ontological constraints (the AGENT of BATHE is, by default, HUMAN, but its basic semantic constraint is ANIMAL) (Nirenburg and McShane 2016b; McShane et al. 2017). If this example seems farfetched, think about how many canonical and slang terms one can use for humans at all stages of life — from *schoolchild* to *oldster* to *dude* to *punk* — any of which could be missing from the agent's lexicon at a given time. The results of this on-the-fly learning can be used immediately and can then either be forgotten, be recorded permanently in the knowledge base, or be passed on for human vetting. If simpler word-learning approaches do not apply, then the agent can leverage the big guns of learning by reading: it can collect contexts that use the word from a large corpus, classify those contexts into hypothesized senses of the word, and then attempt to determine the meaning of the word in each of those classes of examples (ibid). Although this kind of learning is the Holy Grail of knowledge-based systems, it does not, as yet, yield high-quality results. Other options for handling unknown words include asking a human to define them, ignoring them, or waiting to see if further context makes their meaning clear.

Seeking Indirect Speech Act Interpretations.

Many utterances can have both direct and indirect meanings. To take a typical example, *I'm cold* directly

reports my feeling of chilliness but it can indirectly imply anything from “Close the window behind you” to “Do something about it, I don't care what” to “Although I know you can't do anything to change this state of affairs, please say something sympathetic or encouraging.” Since indirect speech acts are most often requests or commands expressed as interrogative or declarative statements (*Would you mind helping me fold the tablecloth?; I'll never finish on time by myself; The lawn needs mowing*), every declarative or interrogative statement the agent receives could, in principle, be a cloaked indirect speech act. This could send the agent on a never-ending wild goose chase for indirect-speech-act interpretations:

Ann: “I love chocolate!”

LEIA (*thinking*): What does she want me to do about that?

The question is, how to constrain this chase? One way is to record in the lexicon a large inventory of canonical patterns for expressing indirect speech acts. For example, among the dozens of frequent paraphrases for asking someone to do X are *I'd like to ask you to do X*, *Would you mind doing X?*, *It would be (really) great if you'd do X*, and so on. Since formulas like these are known by people, we must make them known to our agents as well, thus taking care of a large number of indirect speech situations. A more domain-specific way of detecting indirect speech acts relies on domain modeling, as described earlier on the example of a robotic helper for a surgeon. As long as the agent understands the inventory of actions it can take, it can attempt to align any speech act with one of those actions. However, as mentioned earlier, seeking indirect speech act interpretations too rigorously will clearly be counterproductive: after all, many utterances are simply for information, with the interlocutor not expected to respond at all.

Managing Residual Ambiguity or Incongruity

In many cases, the agent will either not be able to disambiguate some aspect of input (residual ambiguity) or it will have no high-confidence analyses available (incongruity) (for discussion of both of these, see Nirenburg and McShane 2016b). The question then becomes, Does this matter? That depends on the application. On the lenient end of the spectrum is the Senior Companion system (Wilks et al. 2011), which seeks to keep human users engaged in conversation while reminiscing over photographs. This brilliantly selected domain is at once forgiving of near-term limitations in language processing and AI, and open to iterative enhancements over time. By contrast, a language-endowed military robot capable of carrying out irreversible actions offers no tolerance for less than full and confident language understanding as soon as it is deployed.¹⁴

Pursuing Implicatures

A longstanding and largely unsolvable philosophical and practical problem is, Where does language understanding end and general reasoning begin? Given a dialogue exchange like (9), what should the text meaning representation include?¹⁵

- (9) A: Gerard is gluten-free.
B: But he eats pizza.

Clearly, there are two assertions: *Gerard does not eat food containing gluten* and *Gerard eats pizza*. The “but” signals a contrast between them. However, the specific implicatures of B’s statement, and the type of contrast involved, depend upon such things as A’s and B’s individual and shared knowledge, their attitude toward Gerard, B’s intonation, and their goals in having this conversation to begin with. For example, does B know that gluten-free pizza exists? Is he saying that has seen Gerard eat regular, gluten-containing pizza? Is he confused by an apparent incongruity or is he trying to discredit Gerard or speaker A? Making all of these determinations depends not only on the larger speech context, but also on the successful mindreading (that is, mental model ascription) of all participants.

Although we have, in fact, pursued mindreading in OntoAgent (for example, McShane, Nirenburg, and Jarrell 2013;), describing those strategies goes beyond the scope of this article. I mention the issue of implicatures only to underscore that, when we evaluate agents for their ability to understand language, we must be very clear about what we mean. I couldn’t agree more with Ray Jackendoff’s opinion that we cannot, as linguists, draw a tight circle around linguistic meaning and expect all other aspects of meaning to be taken care of by someone else. He writes:

If linguists don’t do it [deal with the complexity of world knowledge and how language connects with perception], it isn’t as if psychologists are going to step in and take care of it for us. At the moment, only linguists (and to some extent philosophers) have any grasp of the complexity of meaning; in all the other disciplines, meaning is reduced at best to a toy system, often lacking structure altogether. Naturally, it’s daunting to take on a problem of this size. But the potential rewards are great: if anything in linguistics is the holy grail, the key to human nature, this is it. (Jackendoff 2007, p. 257)

Although pursuing this level of understanding is both correct and necessary, the practical down side is clear: the higher the language understanding bar, the more difficult it is to show results that will be positively evaluated in the current climate, which so strongly favors breadth over depth and immediate results over long-term prospects.

Conclusion

Much of this discussion has concentrated on ways in which knowledge and reasoning can be leveraged for

language understanding. However, I have concluded with an example in which basic language understanding serves as a springboard for reasoning about things other than language, such as the knowledge, opinions, and goals of one’s interlocutor. This type of reasoning falls between the cracks of agent modeling communities, equally far from raw perception as it is from the flavors of reasoning pursued by formal semanticists and logicians. The need for agents to engage in this type of holistic modeling argues in favor of integrated architectures like OntoAgent.

Let us now briefly return the status of LEIA implementations, with details available in the cited references. The two main implementations of the OS approach to language understanding, called OntoSem and OntoSem2, use the same inventory of microtheories and knowledge bases, but whereas OntoSem interpreted sentences on the whole, OntoSem2 works incrementally (McShane, Nirenburg, and Beale 2016; McShane and Nirenburg 2016). As expected for a research and development effort, more functionalities have been specified than have been implemented, and more have been implemented than have been formally evaluated. Formal evaluations have so far targeted word sense disambiguation (McShane, Nirenburg, and Beale 2016), multiword expression processing (McShane, Nirenburg, and Beale 2015), the processing of difficult referring expressions (McShane and Babkin 2016b), and the detection and resolution of VP ellipsis (McShane and Babkin 2016a). Algorithms that have been implemented¹⁶ but have not been formally evaluated address nominal compounding (McShane, Beale, and Babkin 2014) and fragment interpretation (McShane, Nirenburg, and Beale 2005). We are currently preparing for the first formal evaluation of OntoSem2, which has two foci: threading word sense disambiguation with reference resolution (McShane 2009; McShane and Nirenburg 2013; McShane, Beale, and Nirenburg 2010), and determining the extent to which local dependencies can and cannot resolve lexical ambiguity. The virtual patients in the MVP system (cited earlier) were able to learn new words, concepts, and properties of concepts, and they could interpret various types of indirect and elliptical utterances. Approaches to incorporating mindreading and reasoning with language understanding are detailed in the papers by McShane et al. (2012); McShane, Nirenburg, and Jarrell (2013); McShane (2014); McShane and Nirenburg (2015); and Nirenburg and McShane (2015). Our group’s near-term agenda includes endowing a furniture-building robot with language understanding capabilities (Roncone, Mangin, and Scassellati 2017) and configuring a dialogue agent that is not only able to compute meaning incrementally (albeit with human-like levels of residual ambiguity) but also to recover from unexpected input using expectation-driven strategies that leverage domain knowledge and mindreading of the inter-

locutor (Nirenburg and McShane 2016b; McShane, Blissett, and Nirenburg 2017).

How long will it take for LEIA applications to come to fruition? About the same amount of time as it takes to learn to play the violin: anywhere from a year to a lifetime, depending on time spent, resources available, and target quality and coverage.

Acknowledgments

This research was supported in part by Grant #N00014-16-1-2118 from the US Office of Naval Research. Any opinions or findings expressed in this material are those of the author and do not necessarily reflect the views of the Office of Naval Research. Thanks to Micah Clark for his insightful comments on a draft of this article.

Notes

1. Church (2011, p. 18) writes: “[John] Pierce objects to attempts to sell science as something other than it is (for example, applications), as well as attempts to misrepresent progress with misleading demos and/or mindless metrics (such as the kinds of evaluations that are routinely performed today).” Evaluating NLU systems using the metrics imposed by the NLP community is not only impossible, it falls squarely into the category of mindless metrics.
2. Cited from en.wikipedia.org/wiki/Detroit on December 10, 2016.
3. Some annotations have been removed for concise presentation.
4. It does, however, follow a typical NLP practice of isolating a particular phenomenon, such as reference resolution or word-sense disambiguation, annotating a corpus for a subset of its realizations, then staging a competition among machine-learning systems trained using that corpus. The utility of these task-specific competitions, whose results, to my knowledge, are rarely incorporated into application systems, remains to be seen.
5. The reasoning strategies presented here are selective, not comprehensive. For discussion of yet another reasoning strategy — reasoning by analogy — see Forbus and Hinrichs (2017).
6. The OS ontology is language independent. The names of concepts look like English words only for the benefit of the humans who acquire the knowledge resources and test and evaluate the system. For the system’s purposes, concept names could as easily be randomly selected sequences of characters.
7. There have been two main implementations of the theory of OS. Implementations prior to 2015 carried out sentence-level analysis, meaning that they considered whole sentences at once, an approach typical for syntactic parsers. By contrast, the implementation under development since 2015 pursues incremental (word-by-word) analysis, which more closely emulates what people do. See McShane and Nirenburg (2016) for an in-depth juxtaposition of these implementations.
8. We currently use select outputs from the Stanford CoreNLP tool set (Manning et al. 2014).
9. A more complete lexicon would include many more

phrasal senses, such as *eat one’s hat*, *eat one’s heart out*, *eat someone alive*, and so on.

10. Yes, a lion can eat a human ... and a car can be hot pink, and some dogs have no tails — none of which is covered by our current ontology, which is intended to provide agents with knowledge of how the world typically works.

11. [e] indicates an empty category — that is, ellipsis. The italics indicate the antecedent.

12. Although this might seem like the obvious approach, it is actually not typical in mainstream NLP, where entities of interest in a corpus — so-called markables — tend to be selected manually prior to system training and evaluation (for the example of coreference resolution, see Hirschman and Chinchor [1997]).

13. See Tanenhaus et al. (1995) for a discussion of grounding linguistic references in the real world as early as possible.

14. Of course, full language understanding far from exhausts such a robot’s responsibilities, as discussed by Matthias Scheutz (Scheutz 2017) with respect to ethics.

15. This example arose in conversation with Selmer Bringsjord about the division of labor between language processing and general reasoning.

16. In some cases implementations are partial. Most microtheories include aspects ranging from simple to extremely complex. Consider the example of nominal compounds: in the simplest case, they are recorded explicitly in the lexicon; in the most complex case, the agent doesn’t know the meaning of either of the words; and there are many eventualities in between. In developing microtheories, we attempt to flesh out the full problem space even if we cannot immediately achieve high-quality results for the more difficult component phenomena.

References

- Church, K. 2011. A Pendulum Swung Too Far. *Linguistic Issues in Language Technology* 6(5): 1–27.
- Du Bois, J. W.; Chafe, W. L.; Meyer, C.; Thompson S. A.; Englebretson, R.; and Martey, N. 2000–2005. *Santa Barbara Corpus of Spoken American English*, Parts 1–4. Philadelphia, PA: Linguistic Data Consortium.
- Forbus, K. D., and Hinrichs, T. R. 2017. Analogy and Relational Representations in the Companion Cognitive Architecture. *AI Magazine* 38(4). doi.org/10.1609/aimag.v38i4.2743
- Graff, D., and Cieri, C. 2003. *English Gigaword*. Philadelphia, PA: Linguistic Data Consortium.
- Hirschman, L., and Chinchor, N. 1997. MUC-7 Coreference Task Definition. Retrieval Group, Informational Access Division, National Institute of Standards and Technology, Version 3. Gaithersburg, MD: US Commerce Department. (www-nlpir.nist.gov/related_projects/muc/proceedings/co_task.html)
- Jackendoff, R. 2007. A Whole Lot of Challenges for Linguists. *Journal of English Linguistics* 35(3): 253–262. doi.org/10.1177/0075424207305598
- Lindes, P., and Laird, J. E. 2016. Toward Integrating Cognitive Linguistics and Cognitive Language Processing. Paper presented at the 14th International Conference on Cognitive Modeling (ICCM). University Park, Pennsylvania, 4–6 August.

- Laird, J.; Lebiere, C.; and Rosenbloom, P. 2017. A Standard Model of the Mind: Toward a Common Computational Framework Across Artificial Intelligence, Cognitive Science, Neuroscience, and Robotics. *AI Magazine* 38(4). doi.org/10.1609/aimag.v38i4.2744
- Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S. J.; and McClosky, D. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 55–60. Stroudsburg, PA: The Association for Computational Linguistics. doi.org/10.3115/v1/P14-5010
- McShane, M. 2009. Reference Resolution Challenges for an Intelligent Agent: The Need for Knowledge. *IEEE Intelligent Systems* 24(4): 47–58. doi.org/10.1109/MIS.2009.79
- McShane, M. 2014. Parameterizing Mental Model Ascription Across Intelligent Agents. *Interaction Studies* 15(3): 404–425.
- McShane, M., and Babkin, P. 2016a. Detection and Resolution of Verb Phrase Ellipsis. *Linguistic Issues in Language Technology* 13(1): 1–34.
- McShane, M., and Babkin, P. 2016b. Resolving Difficult Referring Expressions. *Advances in Cognitive Systems* 4: 247–263.
- McShane, M., and Nirenburg, S. 2012. A Knowledge Representation Language for Natural Language Processing, Simulation, and Reasoning. *International Journal of Semantic Computing* 6(1): 3–23. doi.org/10.1142/S1793351X12400016
- McShane, M., and Nirenburg, S. 2013. Use of Ontology, Lexicon, and Fact Repository for Reference Resolution in Ontological Semantics. In *New Trends of Research in Ontologies and Lexical Resources*, ed. A. Oltramari, P. Vossen, L. Qin, and E. Hovy, 157–185. Berlin: Springer. doi.org/10.1007/978-3-642-31782-8_9
- McShane, M., and Nirenburg, S. 2015. OntoAgents Gauge Their Confidence in Language Understanding. In *Self-Confidence in Autonomous Systems: Papers from the AAAI Fall Symposium*, ed. N. Ahmed, M. Cummings, and C. Mille, 22–29. AAAI Technical Report FS-15-05. Menlo Park, CA: AAAI Press.
- McShane, M.; and Nirenburg, S. 2016. Decisions for Semantic Analysis in Cognitive Systems. In *Proceedings of The Fourth Annual Conference on Advances in Cognitive Systems*. Palo Alto, CA: Cognitive Systems Foundation.
- McShane, M.; Beale, S.; and Babkin, P. 2014. Nominal Compound Interpretation by Intelligent Agents. *Linguistic Issues in Language Technology* 10(1): 1–34.
- McShane, M.; Beale, S.; and Nirenburg, S. 2010. Reference Resolution Supporting Lexical Disambiguation. In *Proceedings of the 2010 IEEE Fourth International Conference on Semantic Computing*, 56–59. Los Alamitos, CA: IEEE Computer Society. doi.org/10.1109/ICSC.2010.103
- McShane, M.; Beale, S.; Nirenburg, S.; Jarrell, B.; and Fantry, G. 2012. Inconsistency as a Diagnostic Tool in a Society of Intelligent Agents. *Artificial Intelligence in Medicine* 55(3): 137–48. doi.org/10.1016/j.artmed.2012.04.005
- McShane, M.; Blissett, K.; and Nirenburg, I. 2017. Treating Unexpected Input in Incremental Semantic Analysis. In *Proceedings of The Fifth Annual Conference on Advances in Cognitive Systems*. Palo Alto, CA: Cognitive Systems Foundation.
- McShane, M.; Nirenburg, S.; and Beale, S. 2005. Semantics-Based Resolution of Fragments and Underspecified Structures. *Traitement Automatique des Langues* 46(1): 163–184.
- McShane, M.; Nirenburg, S.; and Beale, S. 2015. The Ontological Semantic Treatment of Multiword Expressions. *Linguisticæ Investigationes* 38(1): 73–110. John Benjamins Publishing Company. doi.org/10.1075/li.38.1.03mcs
- McShane, M.; Nirenburg, S.; and Beale, S. 2016. Language Understanding with Ontological Semantics. *Advances in Cognitive Systems* 4: 35–55.
- McShane, M.; Nirenburg, S.; and Jarrell, B. 2013. Modeling Decision-Making Biases. *Biologically-Inspired Cognitive Architectures* 3 (January 2013): 39–50. doi.org/10.1016/j.bica.2012.09.001
- Nirenburg, S., and McShane, M. 2015. The Interplay of Language Processing, Reasoning, and Decision-Making in Cognitive Computing. In *Natural Language Processing and Information Systems*, Proceedings of the 20th International Conference on Applications of Natural Language to Information Systems, Springer Lecture Notes in Computer Science 9103, ed. C. Biemann, S. Handschuh, A. Freitas, F. Mezziane, and E. Métais, 167–179. Berlin: Springer.
- Nirenburg, S., and McShane, M. 2016a. Natural Language Processing. In *The Oxford Handbook of Cognitive Science*, Volume 1, ed. S. E. F. Chipman. Oxford, UK: Oxford University Press.
- Nirenburg, S., and McShane, M. 2016b. Slashing Metaphor with Occam's Razor. In *Proceedings of The Fourth Annual Conference on Advances in Cognitive Systems*. Palo Alto, CA: Cognitive Systems Foundation.
- Nirenburg, S., and Raskin, V. 2004. *Ontological Semantics*. Cambridge, MS: The MIT Press.
- Nirenburg, S.; McShane, M.; and Beale, S. 2008. A Simulated Physiological/Cognitive “Double Agent.” In *Naturally Inspired Cognitive Architectures: Papers from the AAAI Fall Symposium*, ed. J. Beal, P. Bello, N. Cassimatis, M. Coen, and P. Winston. AAAI Technical Report FS-08-06. Menlo Park, CA: AAAI Press.
- Roncione, A.; Mangin, O.; and Scassellati, B. 2017. Transparent Role Assignment and Task Allocation in Human Robot Collaboration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2017)*. Piscataway, NJ: Institute of Electrical and Electronics Engineers. doi.org/10.1109/ICRA.2017.7989122
- Schank, R. C., and Abelson, R. P. 1977. *Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures* (Artificial Intelligence Series). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Tanenhaus, M. K.; Spivey-Knowlton, M. J.; Eberhard, K., M.; and Sedivy, J. C. 1995. Integration of Visual and Linguistic Information in Spoken Language Comprehension. *Science* 268(5217)(Jun 16): 1632–1634.
- Wilks, Y. 2000. Is Word Sense Disambiguation Just One More NLP Task? *Computers and the Humanities* 34(1): 235–243. doi.org/10.1023/A:1002656922270
- Wilks, Y.; Catizone, R.; Worgan, S.; Dingli, A.; Moore, R.; Field, D.; and Cheng, W. 2011. A Prototype for a Conversational Companion for Reminiscing About Images. *Computer Speech & Language* 25(2): 140–157. doi.org/10.1016/j.csl.2010.04.002
- Zaenen, A. 2006. Mark-up Barking Up the Wrong Tree. *Computational Linguistics* 32(4): 577–580. doi.org/10.1162/coli.2006.32.4.577

Marjorie McShane is an associate professor of cognitive science, and codirector of the Language-Ended Intelligent Agents (LEIA) lab, at Rensselaer Polytechnic Institute. Her research focuses on enabling multifunctional intelligent agents to perform with the linguistic dexterity of their human collaborators.