Advances in Cognitive Systems 6 (2018) 1–20 Submitted 6/2018; published 8/2018

Multi-Stage Language Understanding and Actionability

Marjorie McShane MARGEMC34@GMAIL.COM
Sergei Nirenburg ZAVEDOMO@GMAIL.COM
Jesse English DRJESSEENGLISH@GMAIL.COM
Cognitive Science Department, Rensselaer Polytechnic Institute, Troy, NY, 12180 USA

Abstract

Natural language understanding (NLU) by language-endowed intelligent agents (LEIAs) is modeled as a multi-stage process that involves extensive reasoning both during and between stages. The LEIA attempts to arrive at an *actionable* interpretation of each input as defined by its plans and goals. The paper describes: a) how NLU is integrated with general reasoning in an agent architecture, b) how language understanding is staged according to depth/precision and complexity of analysis, c) how a large number of difficult linguistic phenomena are treated in a unified framework, and d) how natural language can be approached as it occurs in real life, not sanitized or simplified, and without the unnecessary and unrealistic expectation that every utterance must be – or even, in principle, *can* be – fully understood by the interlocutor. The paper makes descriptive, theoretical, and methodological contributions. It addresses technological issues – including the status of the implementation, system evaluation, and scalability – only in passing, as they involve complex problems that cannot be adequately treated in the available space.

1. Introduction

The problem. Language use by real people in real contexts is not a textbook-like enterprise in which speakers generate exclusively clear, complete and well-formed utterances that are fully and confidently understood by their interlocutors. Instead, real speech is messy, and real interlocutors can lack the background, attention, or interest to make full sense of what is said. Cognitive models of language use must address these realities and focus, like people do, on functional sufficiency – i.e., the ability to communicate and interpret information of mutual interest such that its meaning is *actionable* by the interlocutor.

This paper describes how language-endowed intelligent agents (LEIAs) configured in the OntoAgent cognitive architecture (McShane & Nirenburg, 2012) carry out incremental natural language understanding (NLU) using a multi-stage, reasoning-rich, actionability-oriented approach. A special focus is coverage of linguistic phenomena, which we believe will be useful to the cognitive systems community for two reasons. First, although many cognitive architectures (e.g., Lindes & Laird, 2016; Scheutz et al., 2017) include some natural language processing capabilities, we are not aware of any NLU systems that are as deep and cover the inventory of real language phenomena we address. All developers of NLU capabilities will, at some point, need to grapple with the full scope of challenges natural language imposes, and this paper offers a practical method to attain this ability. Second, cognitive systems developers who have not directly engaged with NLU should find this an accessible introduction to the problem space.

1.1 Theoretical Tenets

NLU by LEIAs follows the theory of Ontological Semantics, as originally described in Nirenburg and Raskin (2004) and enhanced subsequently. Of the many theoretical tenets underpinning the LEIA architecture (addressing language, reasoning, simulation, and learning) the following are particularly relevant for this paper:

1. NLU is responsible for the semantic and pragmatic analysis of language, and it results in unambiguous, fully specified, ontologically-grounded *text meaning representations* (TMRs). The production of TMRs is, we believe, what the reasoning community has been expecting from the language community for the past 60 years, but mainstream NLP has not pursued this goal for over 25 years (Nirenburg & McShane, 2016a).
2. NLU for LEIAs focuses on *actionability*, i.e., determining when an input is understood with sufficient completeness and confidence to support the agent's reasoning about action. This stands in contrast to the non-real-world assumption that every utterance has a perfect, complete, and singular interpretation that must necessarily be fully understood by the interlocutor.
3. NLU can only be successfully achieved within a *comprehensive agent architecture* because NLU is only very partially about the surface strings of language utterances.
4. By default, NLU is carried out *incrementally* – roughly, word-by-word – though some applications may be equally served by a sentence-level approach. Incrementality will ultimately permit agents to display such human-like behaviors as interrupting, asking for clarifications, and beginning to act before an utterance is complete.
5. NLU is organized as a sequence of *stages of processing*, from "surfacy" to deep (i.e., knowledge- and reasoning-heavy). Our current model includes the following seven stages: preprocessing, syntactic analysis, basic semantic analysis, (co)reference resolution, extended semantic analysis, plan- and goal-based reasoning about NLU, and augmenting the ontology and lexicon through learning. In many cases, the earlier stages posit a general interpretation that can be made more specific by additional reasoning during later stages. This staging is psychologically motivated: e.g. we can get a general interpretation of *He ate it* without yet grounding it in a context that determines who ate what.
6. There is a *decision point* after each stage of processing each fragment. The LEIA can decide to (a) act based on its current level of understanding, (b) pursue deeper analysis of the input fragment consumed so far, or (c) consume the next element of input. This models a person's ability to act upon incomplete and/or partially understood inputs.
7. The LEIA's *decision functions* about how to proceed with language understanding rely on some parameters common to all applications and some that are specific to a given application.
8. Coverage of a *large inventory of linguistic phenomena* is centrally in purview.
9. Use of a detailed, *construction-rich lexicon* is key to treating a large scope of phenomena in a streamlined way that avoids the well-known pitfalls of large knowledge-based systems.

1.2 Main Claims

In this paper, we present four main claims. These claims, and our approach to validating each of them, are as follows.

- **Claim 1:** The approach to NLU for LEIAs described here covers a large number of linguistic eventualities in a theoretically-motivated and methodologically sound manner. *Validation*: We will present brief overviews of the types of linguistic phenomena treated at each level of analysis and the nature of their linguistic treatment. Integrating so many phenomena in a single NLU system is a large and novel contribution.
- **Claim 2:** The approach addresses the needs of real (not idealized) language interactions as modeled in a real (not idealized) language processing system. *Validation*: We will emphasize the LEIA's methods of dealing with unexpected linguistic input and focusing its attention on what it *can* understand within its domain of interest.
- **Claim 3:** This approach to NLU is computer tractable. *Validation*: What we describe is under implementation at the time of writing. Cited references report formal evaluations that have been carried out for a subset of the functionalities and describe computer-tractable algorithms for others.
- **Claim 4:** This approach to NLU is useful for LEIAs in applications. *Validation*: The validation for this claim is, as yet, informal and consists of three parts: a) the output of NLU by LEIAs answers the stated needs of the machine reasoning community to convert messy natural language into unambiguous formal constructs; b) LEIAs successfully served as virtual patients in the Maryland Virtual Patient clinician training prototype application (Nirenburg, McShane, & Beale, 2008); and c) LEIAs are currently integrated in a robotic application (Nirenburg et al., 2018).

Rather than devote space to an overview of NLU within Ontological Semantics, which we have already amply described (McShane, Nirenburg & Beale, 2016; Nirenburg & McShane, 2016b; McShane, Blissett & Nirenburg, 2017, among others), we will describe system operation and output in the narrative about each stage of processing.

2. Stage 1: Preprocessing

Preprocessing includes text segmentation, part-of-speech tagging, lexical lookup, and morphological analysis. It is carried out by the Stanford CoreNLP Natural Language Processing Toolkit (Manning et al., 2014). We use an external toolset for preprocessing heuristics – as well as some aspects of syntactic analysis (Stage 2) – because (a) Ontological Semantics makes no theoretical claims about how pre-semantic analysis is carried out, (b) all pre-semantic heuristics are considered overridable evidence by the semantic analyzer, and (c) we choose to devote our resources to semantics and reasoning, which includes recovering from mistakes and insufficiencies of pre-semantic analysis.

Preprocessing is a separate stage because its output signals a decision point for the LEIA For example, if the most recent input element in the incremental analysis is the article 'the' (*He finished the*), the LEIA will not pursue deeper analysis of this fragment but, instead, will directly ask for the next word of input. Similarly, the lexical lookup available after tokenization can be

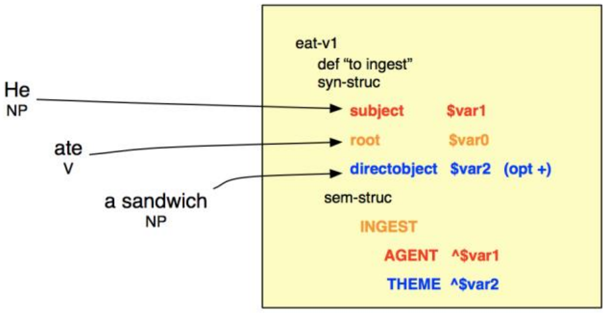

Figure 1. The syntactic mapping of the input string *He ate a sandwich* onto the first verbal sense of *eat, eat-v1*.

sufficient for the LEIA to determine that an input is not relevant to it. This can be useful, e.g., if a LEIA working in a narrow domain is expected to hear a lot of off-topic conversation by its human collaborators, or if a LEIA needs to skim a large volume of text in order to find passages of potential interest. Operationally, this skimming can be done by generating, prior to runtime, the list of words from the lexicon that map to the LEIA's concepts of interest.

3. Stage 2: Syntactic Analysis

Syntactic analysis establishes syntactic dependencies that can suggest corresponding semantic dependencies. Although CoreNLP produces full constituent and dependency parses, we use their results selectively[1] and supplement them with considerable LEIA-specific analysis. In brief, low-level syntactic constituents (e.g., *NP* and *comp* [a clausal verbal complement]) are used as input to the process of syntactic mapping, which links elements of input to constituents in the syn-struc ("syntactic structure") zones of senses in the LEIA's lexicon. Syntactic mapping is a strictly syntax-oriented process that answers the question, "Syntactically speaking, what is the best combination of word senses to cover this input?" Figure 1 illustrates the mapping process for the input *He ate a sandwich*, showing a pretty-printed excerpt from the first verbal sense of *eat*. Later on, the semantic analyzer will determine whether the meanings of these variables ("^" indicates 'the meaning of') are appropriate fillers of the AGENT and THEME case-roles of INGEST.

The details of the syntactic mapping program are not important for our purposes, but what *is* important is that the LEIA is prepared for this process to *not* work out perfectly in every case, and so it has several recovery methods. For example, (a) it treats unknown words by positing a lexical sense that reflects the attested syntactic dependency structure along with an underspecified semantic interpretation that is honed during semantic analysis; (b) it detects syntactic irregularities, such as repetitions and disfluencies, and strips them away before rerunning syntactic analysis to see if the pruned version results in a canonical parse; and (c) it detects the syntactic expectations of certain types of fragments (e.g., quantifiers used without a head, as in

[1] Of particular note is the fact that we do not centrally rely on the CoreNLP's dependency parse for syntactic heuristics since we found it too error-prone, particularly in the genre of informal dialogs.

"How many cookies do you want?" "*Five*.") and inserts a dummy head (here, a noun) that permits the analysis to proceed in the normal way. If these recovery procedures do not result in a canonical syntactic structure, then the basic "syntax informs semantics" method of NLU is circumvented and the agent opts for the method of meaning composition described in Stage 6.

The LEIA's decision-making after syntactic analysis mostly pertains to non-sentence-final fragments, for which the LEIA must decide whether to launch semantic analysis or wait for more words of input. The decision depends upon such things as the nature of the fragment (does it represent a complete syntactic dependency structure?) and the response time requirements of the application domain (if there is no rush, the LEIA can wait until the end of the sentence to carry out semantic analysis).

4. Stage 3: Basic Semantic Analysis

NLU by LEIAs results in what we call text meaning representations, or TMRs. Below is a simplified (i.e., omitting inverses and metadata) TMR for the simple sentence **A brown squirrel is eating a nut**.

```
INGEST-1
   AGENT       SQUIRREL-1
   THEME       NUT-FOODSTUFF-1
   TIME        find-anchor-time
```

```
SQUIRREL-1
   COLOR       brown
   COREF       block-coref
```

```
NUT-FOODSTUFF-1
   COREF       block-coref
```

Generating a TMR requires disambiguating each lexeme – i.e., understanding it as an instance of a particular concept in the ontology – and combining those interpretations into a semantic dependency structure. For the above sentence, the LEIA has to select between three senses of *eat* (INGEST as well as the literal and metaphorical meanings of the phrasal *eat away at*), and three senses of *nut* (the foodstuff, the crazy human, and the machine part). Four of the lexicon senses it uses to create this TMR are shown below using a simplified formalism.

```
eat-v1
   def/ex      "ingest; He was eating (cheese). "
   syn-struc
      subject       $var1 (cat n)
      root          $var0 (cat v)
      directobject  $var2 (cat n)  (opt +)
   sem-struc
      INGEST
           AGENT    ^$var1 (sem ANIMAL)
           THEME    ^$var2 (sem INGESTIBLE)
    meaning-procedure nil
```

```
a-art1
   def/ex      "no coref.; He wants a big house."
   syn-struc
      $var0    (cat art)
      $var1    (cat n)
   sem-struc   nil
   meaning-procedure block-coref
```

```
squirrel-n1
   def/ex     "an animal; A squirrel ran b.y"
   syn-struc
      (root $var0) (cat n)
   sem-struc
      SQUIRREL
   meaning-procedure  nil
```

```
brown-adj1
   def/ex       "a color; brown shoes"
   syn-struc
      $var1 (cat n)
         mods    (root $var0) (cat adj)
   sem-struc
      ^$var1 (sem PHYSICAL-OBJECT)
         COLOR      brown
   meaning-procedure nil
```

Some notes about the content and form of the lexical senses: the entry for 'nut' is analogous to the one for 'squirrel' and is not shown for reasons of space; the semantic constraints in italics are not actually listed in the lexicon, they are accessed from the ontology at runtime; and COLOR is a literal attribute so its fillers, including 'brown', are not concepts – and are, therefore, not written in small caps.

Returning to the TMR, the italicized elements are calls to procedural semantic routines that will be carried out at a later stage of processing. *Find-anchor-time* attempts to establish the time of speech, since this utterance is the present tense. *Block-coref* indicates that coreference resolution should not be run – this is a new instance of the given type of object. Information recorded in the ontology supports the disambiguation: i.e., the 'person' and 'machine part' meanings of 'nut' are not valid THEMEs of INGEST, but the foodstuff is.

Operationally speaking, the TMR for our sentence is generated by: a) copying the sem-struc of *eat-v1* into the nascent text meaning representation; b) translating the concept type (INGEST) into a numbered instance (INGEST-1); and c) replacing the variables (^$var1, ^$var2) with their appropriate interpretations (SQUIRREL-1 [COLOR brown], NUT-FOODSTUFF-1). In terms of runtime reasoning, this example is as simple as it gets since it involves only constraint satisfaction, and all constraints match in a unique and satisfactory way. "Simple" constraint satisfaction does not, however, come for free: its precondition is the availability of a high-quality lexicon (currently containing around 30,000 senses) and ontology (currently containing around 9,000 property-rich concepts). Candidate TMRs are scored based on how closely the syntactic and semantic expectations of lexical senses are satisfied.

This stage of processing covers not only simple inputs like our example; it provides a *basic* level of analysis for many linguistic phenomena that are typically regarded as advanced. This is possible thanks to the LEIA's construction-rich, phrase-rich lexicon that *anticipates* these phenomena. Creating such a lexicon is not merely an engineering solution, it is a theoretically-grounded aspect of cognitive modeling on two counts. First, native speakers of a language store knowledge not only of individual words, but also of how the words are used together in idiosyncratic ways. Second, many inputs actually do license multiple levels of interpretation. "He ate it", "I can't!" and "Yours is better" make some sense even before we know *who* ate *what, what* can't you do? and *what* is better than *what?*

For reasons of space, we will not describe all phenomena covered at this stage of analysis (which include the treatment of modification, modality, aspect, sets, etc.) but, instead, will concentrate on those phenomena that are traditionally considered more difficult – and are often treated as separate, free-standing research topics. We will often make a distinction between how different *instances* of a phenomenon are treated: e.g., some might be detected and fully interpreted at this stage, whereas others are detected and resolved in an underspecified way, awaiting contextual grounding at a later stage.

1. **Referring expressions** are analyzed at this stage using a lexically specified static representation (e.g., *he* is described as HUMAN (GENDER male)) that is supplied with a call to a procedural semantic routine which, if launched during Stage 4, will attempt to resolve the reference contextually. The call to the procedural semantic routine is recorded in the *meaning-procedures* zone of associated lexical senses (cf. 'a-art1' in Table 1) and fills the COREF slot in the basic TMR.

2. **Multiword expressions** of any syntactic form, and with any combination of fixed and variable elements, are recorded in the lexicon and processed as a matter of course by the semantic analyzer (McShane, Nirenburg, & Beale, 2015). We use multiword expressions for many purposes: for idioms, common collocations, and even entire utterances if their form-to-meaning mapping cannot reliably be computed using the currently available general procedures. As an example of the latter, assume that a robot communicating with a particular user in a particular situation needs to understand that the highly elliptical *"Hey, no, a Phillips"* means "Do not give me what you have just picked up, give me a Phillips screwdriver instead". We can record this whole utterance as a phrasal lexicon entry with the intended meaning – as if it were an idiom. Over time, as the LEIA becomes more adept at reasoning about ellipsis, it will be able to compute such meanings productively – the methods for attaining this capability are under development. The point is that creating such phrasal lexicon entries requires no new representational methods and ensures that particularly difficult language problems do not impede the overall development of the LEIA's many integrated capabilities covering perception, reasoning and action.

3. Many **indirect speech acts** follow well-known lexical patterns. For example, *I need to know your address* and *You need to tell me your address* are both ways of asking for the interlocutor's address. Of course, every formula that can serve as an indirect speech act also offers a direct interpretation: e.g., *I need to know your address, but since you've already refused to tell me, I'll look it up in your records.* Therefore, the LEIA treats phrasal lexicon entries representing indirect speech acts the same way as all other phrasal entries: the phrasal interpretation receives a higher score than compositional interpretations, but the latter are available if the phrasal interpretation fails for some pragmatic reason – something that will be evaluated by the LEIA at Stage 6.

4. **Nominal compounds** (NNs) are sequences of two or more nouns in which one modifies the other(s): e.g., *glass bank.* The full analysis of a NN involves both contextually disambiguating the nouns (something not undertaken by mainstream NLP) and establishing the semantic relationship between them. As reported in McShane, Beale, and Babkin (2014), different types of NNs are treated in different ways at different stages of processing. At this stage of processing, three things happen. (i) Lexically recorded NNs, such as *attorney general* and *drug trial* are analyzed as multiword expressions. (ii) Lexically-anchored NN patterns that include ontologically-constrained variables are treated. For example, one sense of the word 'fishing' expects an NN whose first N is semantically constrained to a type of FISH. The meaning of the NN is listed as FISHING-EVENT (THEME [*the given type of fish*]). So, this sense permits 'salmon

fishing' to be confidently analyzed as FISHING-EVENT (THEME SALMON). (iii) NNs that are not covered by these lexicalized methods are analyzed at this stage using a generic RELATION and supplied with a call to a procedural semantic routine (recorded as metadata in the TMR) that will attempt to further specify the nature of that RELATION at Stage 5.

5. Many **metaphors** are conventional and, as such, their meanings must be recorded in the lexicon – e.g., *fly* can mean *move very quickly* (MOTION-EVENT (SPEED > 0.9[2])), and *attack someone* can mean *criticize intensely* (CRITICIZE (INTENSITY > 0.8)). For LEIAs, the etymological source of such metaphorical meanings is not relevant: after all, it is not necessary for flying to be fast, and one cannot explain why *attack* can mean 'criticize' but the semantically related *assault* cannot. Among non-conventional metaphors, those that are presented in a copular construction – such as Lackoff and Johnson's "*Love is a collaborative work of art*" (1980, p. 139) – are analyzed using a lexical sense of the verb *be* that simply establishes a semantic juxtaposition. This lexical sense is appended with a procedural semantic routine (stored as metadata in the TMR) which, if run at Stage 5, will attempt to determine the salient properties of the predicate nominal (a difficult task, indeed).

6. Like metaphors, many word senses that can historically be analyzed as **metonymies** are, in synchronic terms, regular lexical senses that are recorded in the LEIA's lexicon: e.g., *get a pink slip* (get fired), *red tape* (excessive bureaucratic requirements), *give someone lip* (talk back rudely), *hired gun* (assassin). There are also ontological classes of metonymies: e.g., a piece of clothing can be used for the person wearing it (*Give the red shirt a glass of milk.*). These metonymies are *not* treated at this stage: instead, they result in an incongruency during basic semantic analysis, which results in a low-scoring TMR. This low score is a flag to track down the source of the incongruity at Stage 5.

The next four phenomena relate to ellipsis – a widespread phenomenon in realistic language communication but at, the same time, one of the least pursued phenomena in current NLP due to the absence of textual strings for corpus-based systems to manipulate. A detail about ellipsis that might be overlooked, if not overtly stated, is that the lexical senses used to treat ellipsis support both its *detection* and its *resolution* (in full or in part), with both these tasks being equally difficult.

7. **Verb phrase (VP) ellipsis** is the omission of a VP as licensed by a modal or aspectual verb: *"I can't __!" "But you must __!"* LEIAs detect VP ellipsis using a special sense of each modal and aspectual verb that a) *expects* the VP to be elided (i.e., its syn-struc zone lists just a subject and the modal/aspectual verb), b) posits a generic EVENT in the semantic interpretation (i.e., analyzes "I can't" as if it were "I can't *do-it*"), c) uses that EVENT in the semantic dependency structure of the clause (i.e., makes the meaning of the subject the AGENT of the EVENT), and d) includes a call to a procedural semantic routine (recorded as metadata in the TMR) which, at Stage 4, will attempt to resolve the reference of the EVENT.

8. One subclass of VP ellipsis that is relatively easier to treat involves **VP ellipsis configurations**, which can be recorded as constructions in the lexicon. E.g., *$Subj_1$ V as ADV as $Subj_1$ can/could* [Winny [$_{\text{VP}}$ ran [$_{\text{ADVERBIAL}}$ as fast as she [$_{\text{VP}}$ could __]]]]. These are a boon for processing because the sponsor for the elided VP is in the construction itself, meaning that the

[2] Values of scalar attributes can be actual numbers or points on the abstract {0,1} scale.

entire meaning can be resolved at this stage. Deciding how many such constructions to record, and the balance of literal and variable elements in each one, represents a microcosm of knowledge engineering overall.

9. When aspectual verbs (e.g., *start*) take an NP complement that refers to an ontological OBJECT (e.g., *He is starting a book*), there is always an elided EVENT.[3] This **aspectual + NP$_{\text{OBJECT}}$ ellipsis** is treated using a lexical sense of each aspectual verb that expects an OBJECT complement. At this stage of processing, a generic EVENT is posited and used in the dependency structure. For 'He is starting a book' the TMR will be (EVENT-1 (AGENT HUMAN-1 (GENDER male)) (THEME BOOK-DOCUMENT-1) (TIME *find-anchor-time*). The associated lexical senses include a call to a procedural semantic routine that will attempt to further specify the meaning of the EVENT during Stage 4 of processing.

10. Certain other verbs, when used with certain kinds of OBJECTS, can idiosyncratically predict the ellipsis of a particular type of event: e.g., forgetting something that can be carried means forgetting to BRING that thing. These cases of **idiosyncratic event ellipsis** are recorded in lexical senses and are fully resolved at this stage of processing.

11. **Gapping** is a type of verbal ellipsis that occurs in syntactically parallel configurations such as *Pedro ordered linguini and Charlie ___, spaghetti*. Gapping is not used widely in English. Still, we cover the most common gapping patterns by recording constructions as lexical senses of the only two conjunctions that permit gapping in English: *and* and *but.* Formalism aside, the syntactic descriptions are of the type [Subj V DO *and* Subj2, DO2], and their associated semantic representations reconstruct the elided verb using a different instance of the same EVENT indicated by first-clause verb. As such, gapping is both detected and fully resolved at this stage of processing.

12. As mentioned earlier, **unknown words** are initially treated during syntactic analysis by positing an underspecified lexical sense that reflects the syntactic dependency structure of the input sentence. The linked semantic description analyzes the verb as an EVENT and posits two generic CASE-ROLEs to accommodate the meaning of the subject and direct object. At this stage, the LEIA attempts to narrow down the meaning of the EVENT, as well as select the correct case-roles (e.g., AGENT and THEME) based on the meanings of the arguments using ontological search. This process is described in McShane, Blissett, and Nirenburg (2017).

This ends our brief overviews of some of the more interesting linguistic phenomena subsumed under basic semantic analysis. We have already described the theoretical justification for considering so many phenomena to be part of "basic" analysis, but there is also an important benefit in terms of system building. Large knowledge-based systems become unwieldy if they do not maximally exploit generalized functions and keep knowledge elements tightly organized. We cannot afford to have hundreds of phenomenon-specific functions scattered throughout the analyzer code to deal with each "difficult" linguistic phenomenon. The way we have organized it, the procedures to deal with these phenomena are anchored to lexical senses that give rise to them, and all traces of all calls to procedural semantic routines are recorded explicitly in the metadata of TMRs in support of testing, debugging, and continuous system development.

[3] Note that if the NP object refers to an EVENT, there is no ellipsis. If one *starts an argument*, then the aspect (phase: begin) scopes over the ARGUE event instantiated by the noun 'argument'.

In some cases, basic semantic analysis is sufficient as an end stage of NLU, as for inputs that have no coreference needs and for which the LEIA arrives at a single, high-confidence interpretation (e.g., *A brown squirrel is eating a nut*). In other cases, basic semantic analysis is sufficient to convince the agent that it does not need to analyze the input any more deeply – e.g., if the basic TMR lacks any concepts within the LEIA's domain of interest (something that can be more confidently judged at this stage than it was using the skimming method following preprocessing). But in most cases, the basic TMR serves as input to the more sophisticated reasoning needed to arrive at a full and confident, contextually-grounded interpretation.

The next two stages of analysis, reference resolution (Stage 4) and extended semantic analysis (Stage 5), further specify and/or disambiguate the basic TMR. These stages are similar in that they are still primarily linguistic and domain-independent, in contrast to the plan-and-goal-based methods that will be leveraged, if needed, in Stage 6.

5. Stage 4: Reference Resolution

At this point, all referring expressions have already been detected and provided with a basic semantic analysis. What remains is to ground them in the discourse context. In some cases, this requires textual coreference; in others, it requires coreference with something in the real-world environment that is not mentioned in the linguistic context; and in still others, it requires recognizing that the referring expression is new to the discourse. No matter which of these eventualities obtains, the LEIA must ultimately anchor each referring expression in its memory, which is the *actual* definition of "reference resolution".

The number of programs, rule sets, and reasoning functions that LEIAs bring to bear for reference resolution is too large to even cursorily treat in this space (for an overview of phenomena, see McShane 2009; for aspects of our past work on it within this architecture, see McShane, 2015, and McShane & Babkin, 2016). To give just a taste of the scope of work, consider VP ellipsis. It will have been detected during basic semantic analysis due to the use of a modal or aspectual lacking an EVENT to scope over. If the ellipsis occurs in a VP ellipsis configuration, full resolution will also have been accomplished; but in all other cases, an EVENT will have been posited as an underspecified placeholder that now requires context-based specification.

However, this specification is not limited to pointing to an antecedent in the text. Instead, five semantic determinations must be made, as can be illustrated by the sentence **Mark managed to hand in his project before class yesterday, and Allie did __ today.** (1) What is the verbal/EVENT head of the sponsor? Hand in/SUBMIT. (2) Are the sponsor and elided event in a type-coreference or instance-coreference relationship? Type-coreference (there are 2 different instances of SUBMIT). (3) Do the internal arguments have strict or sloppy coreference (i.e., same or different referents)? Sloppy coreference (there are two different projects). (4) Are modifiers copied or not copied into the resolution? *Before class* is copied but *yesterday* is not! (5) Are modal and/or aspectual meanings in the sponsor clause copied or not copied into the resolution? The modal meaning indicating epiteuctic modality (i.e., 'managed to') is copied.

Of course, citing a simple example like the one above masks the true complexity of the task at hand, as illustrated by sentences like, *The former Massachusetts governor called on United 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]*. A key modeling strategy (for details, see McShane & Nirenburg, in preparation) is to enable LEIAs to determine which examples they can treat with high confidence and which they cannot. In the latter case, they defer reference resolution until they can leverage script-based reasoning or ask a human collaborator for clarification.

After reference decisions are posted, the LEIA can update the preference scores of all candidate TMRs. Consider, for example, the sentence *John's father talked at length with the surgeon and then he promptly started the operation*. Basic coreference procedures for 'he', which rely on lexico-syntactic heuristics, will incorrectly link *he* to *John's father*. It is world knowledge that tells us that surgeons operate. But note that the choice space for the LEIA is actually even more complex than that, since two senses of 'operate' are available in this context: SURGERY and OPERATE-MACHINERY. Whereas both a surgeon and a father are equally fitting as AGENTs of OPERATE-MACHINERY (being HUMANs, they both fit the 'sem' constraint in the ontology), SURGEON is a particularly good fit for the AGENT of SURGERY (it fits the 'default' constraint), whereas FATHER is a poor one (it fits only the 'relaxable-to' constraint). The scoring function at this stage takes into account the results of both lexical disambiguation and reference resolution. Residual referential ambiguities are addressed in Stage 6, using plan- and goal-based reasoning.

6. Stage 5: Extended Semantic Analysis

This stage, like the previous one, a) addresses outstanding cases of ambiguity and underspecification identified during basic semantic analysis, and b) uses linguistically informed, generalized analysis methods – not yet reasoning about plans and goals. Analysis procedures are triggered by the four eventualities described below.

1. All calls to non-reference-oriented **procedural semantic routines,** which were recorded as metadata in the basic TMR, are now run. These seek to further concretize underspecified analyses such as nominal compounds that were initially analyzed using a generic RELATION and unanchored comparisons (*My bike is faster*). For each of these, additional knowledge sources and reasoning functions are brought to bear. Consider the example of nominal compounds.[4] The LEIA consults a special-purpose knowledge base that records prototypical relationships between concepts. For example, TEMPORAL-UNIT + EVENT means that the event occurs at the given time, so *Tuesday flight* is analyzed as FLY-EVENT (TIME TUESDAY). Similar analyses apply to *morning meeting, weekend getaway, 8:00 appointment.* If the given NN is not covered by a recorded ontological pattern, the LEIA uses a shortest-path search through the ontology to hypothesize the most likely relation. If this fails to result in a strong candidate, the relation between the nouns' meanings remains the generic RELATION.

2. Cases of **residual lexical ambiguity** – which are detected by the availability of multiple high-scoring TMR candidates – are addressed by attempting to establish an ontological context for the utterance. For example, in "The police arrived at the port before dawn and arrested the pirates," the key to disambiguating which kind of pirate (PIRATE-AT-SEA or INTELLECTUAL-PROPERTY-THIEF) is found in the preceding clause, not in the local dependency structure. We

[4] Whether deeper NN analysis is subsumed under basic semantic analysis or postponed until this stage is an engineering decision since, in terms of cognitive-modeling, a case can be made for both options.

have developed several methods of recording and reasoning over ontological knowledge in support of such disambiguation. For example, the LEIA searches the ontology to see if any of the meanings of objects mentioned in the immediately preceding context serve as case-role fillers for any EVENTs for which either PIRATE-AT-SEA or INTELLECTUAL-PROPERTY-THIEF is also a case-role filler. If, e.g., the ontology contains WATER-TRAVEL-EVENT with property descriptions that include "AGENT default SAILOR, PIRATE-AT-SEA" and "DESTINATION sem PORT, GEOGRPAHIC-ENTITY", the LEIA hypothesizes that the text is likely about a WATER-TRAVEL-EVENT and the meaning of 'pirate' is PIRATE-AT-SEA. This, and other methods like it, represent knowledge-based approaches to operationalizing the oft-mentioned advice to "just use the context!"

3. Semantic **incongruity** is detected by the absence of any high-scoring TMRs – i.e., the semantic expectations about the correlation between argument-taking heads and their dependents cannot be fulfilled. Sample phenomena that lead to incongruities are non-lexicalized metonymies (*I bought an Audi*) and indirect modifications (*What ensued was a bloodthirsty chase*). Non-lexicalized metonymies are analyzed using a specially developed repository of typical metonymic relationships formulated in terms of ontological concepts. It includes such correspondences as producer for product (*We bought a Toyota*), social group for its representative(s) (*The ASPCA reported…*), and clothing or body part for the person associated with it (*The red hat <big nose> just bumped into a tree*). Replacing the metonymy with its implied class results in the satisfaction of previously unsatisfied selectional constraints. Indirect modifications are handled by anticipatory rules that cover typical eventualities. Taking the example *bloodthirsty chase*, the adjective *bloodthirsty*, when modifying an EVENT, typically describes the animal that is the AGENT of that EVENT. Lexically-linked rules anticipate this type of indirect modification and create the necessary semantic expansion.

4. At this stage of analysis, the LEIA can also fully interpret sentence-level **fragments** that either fulfill a need reflected in the preceding TMR (as in question-answer contexts) or add additional modification to an utterance (e.g., "Get me a coffee. Fast.") For the Q/A contexts, the agent reconstructs the answer as a full meaning representation: "How many cookies do you want?" "Five." → "I want five cookies". For the case of modification, the meaning of the modifier is appended to a copy of the preceding TMR. This subset of fragments can be fully interpreted at this stage because the analysis relies on purely linguistic generalizations, not those requiring plan- and goal-based reasoning. The latter (see Stage 7) is needed for the LEIA to understand that a surgeon saying "Scalpel!" means "Give me a scalpel!"

An interim summary: Let us recap the last three stages. Basic semantic analysis (Stage 3) carries out lexical disambiguation and creates the semantic dependency structure. It covers a large inventory of advanced linguistic phenomena thanks to fine-grained, anticipatory lexical acquisition, but it often results in residual ambiguities (multiple high-scoring TMRs), incongruities (no high-scoring TMRs), and/or underspecifications (as in cases of ellipsis). The next two stages – reference resolution (Stage 4) and extended semantic analysis (Stage 5) – attempt to further specify and contextually ground specific aspects of the TMR using linguistically-oriented knowledge and reasoning that are broadly applicable across agents and applications. In essence, processing through Stage 5 represents what an agent can bring to bear in terms of basic linguistic and world knowledge. In this sense, the end of Stage 5 might be considered a practical "full stop" in the process of NLU. This is important for the following reason: although agents are *able* to engage in incremental understanding, and although they are *able* to make decisions about what to do after each stage of analyzing each fragment, this degree of splitting the NLU process is not necessary for all applications. An alternative deployment strategy for non-urgent interactions is for agents to wait until the end of each utterance to process it, and then to process it through Stage 5 before pausing to decide what to do next. For example, it will, by default, prefer indirect-speech-act interpretations of utterances over informational ones if responding to the indirect speech act is within its capabilities: e.g., given *I need a hammer,* it will retrieve one rather than simply remember the information that the speaker is in need of a hammer. Similarly, it will not further pursue residual ambiguities if all the available interpretations are outside of its scope of interest: "do nothing" is as much an action as any other. However, if reasoning about action is *still* not possible – for reasons described in the next section – then plan- and goal-based reasoning is triggered.

7. Stage 6: Plan- and Goal-Based Reasoning about Language

We will only give a taste of this module because it is difficult to describe its content without introducing a particular agent operating in a particular application. We will report on this component of the system in detail in the future. This taste will involve the five eventualities listed below. In reading these, remember that we are talking about plan- and goal-based reasoning *in service of NLU*: this certainly does *not* exhaust agent reasoning about action overall. In addition, although we repeatedly refer to scripts (complex events in the ontology), we mean scripts that are instantiated in the application – i.e., relevant to the agent's plans and goals – not the full inventory of scripts in the agent's ontology.

1. **Script-based lexical disambiguation**. In some cases, residual lexical and referential ambiguities can be resolved by preferring word meanings that are used in activated scripts. For example, given the input *I like chairs!* there is no linguistic reason to prefer one analysis of 'chair' (CHAIR-FURNITURE) over the other (CHAIRPERSON). However, in a chair-building application, the agent will have a clear, script-based preference: CHAIR-FURNITURE is the only one mentioned in the script. Note that one can also configure LEIAs to use script-based disambiguation preferences *earlier* in the process of NLU, during basic semantic analysis. This is a design choice that could affect efficiency, particularly in time-sensitive applications.

2. **Script-based reference resolution.** If the basic reference resolution fails to confidently identify the sponsor for a referring expression – particularly a personal pronoun, a broad referring expression (such as pronominal 'this' or 'that'), or an elided expression – script-based reasoning is triggered. For example, if a chair-building agent cannot decide, based on the linguistic context, if 'it' in 'Hit it with hammer' is the back of the chair or the nail, it can check its activated scripts and see what the THEME of HIT can be. It should find that NAILs are hit with a HAMMER but CHAIR-BACKs are not.

3. **Script-based analysis of "bag of words/constituents" inputs.** Some inputs are so syntactically irregular that the LEIA will not be able to clean them up and they will not pass through the normal "syntax informs semantics" analysis process. In such cases, the LEIA attempts to cobble together the most probable meaning by largely ignoring syntax (apart from NP chunking) and attempting to piece together the available concept mappings into ontologically

valid dependencies. Orienting around the active script(s) can guide both in lexical disambiguation and in determining the dependency structure.

4. **Detection of linguistically unsignalled indirect speech acts.** Unfortunately for agent-system developers, any statement can effectively serve as a request or command: *I'm bored. The dog hasn't been out in a while.* Determining which statements do and do not involve implicatures that the agent can act upon is a difficult problem that counts among our specific research objectives.

5. **Plan- and goal-oriented analysis of fragments.** Some fragments, particularly bare noun phrases, have not yet been incorporated into the contextual interpretation: e.g., *Scalpel! Two lattes, no sugar.* One generalization is that when a noun phrase functions as an independent utterance that does *not* fulfill the pragmatic need of a previous utterance (as by providing the answer to a question), it tends to mean "Give me [that object]". This is just the tip of the iceberg in incorporating such fragments into the contextual interpretation – a topic we plan to explore bottom-up, using specific dialogs in specific agent applications.

8. Stage 7: Learning Lexicon and Ontology

We mention the agent's learning of lexicon and ontology only in passing, for the sake of completeness, as it is large topic. We have already seen how LEIAs can learn underspecified lexical senses for new words during the basic NLU process. They can also learn new words, concepts, and even scripts in both within the scope of an application and offline. We have explored three learning methods in past work: learning new words by reading (English & Nirenburg, 2010); learning new words and ontological concepts by being told, a capability used by virtual patients in the Maryland Virtual Patient application (Nirenburg, McShane, & Beale, 2008); and learning ontological scripts through language interaction while carrying out a task (Nirenburg et al., 2018). The results of such learning serve to improve future language understanding in a paradigm of lifelong learning by bootstrapping.

9. In the Context of the Field

The necessarily selective comparisons presented here will address potential points of intersection between our work and other cognitive systems research having different emphases.[5] The most obvious dovetailing is with programs of work that concentrate on reasoning and/or robotics and deal with the natural language understanding problem either by avoiding it – i.e., using as inputs handcrafted expressions in a formal metalanguage – or by making progress in areas that allow for bypassing NLU as a central research issue.

Consider three reasoning-centric programs of work for which inputs are typically handcrafted, with developers expecting the natural language processing community to provide

[5] We will not offer head-to-head comparisons with other systems and environments that may have some overlapping methods or goals (e.g., the knowledge bases and software produced by Cycorp, or work within the TRIPS architecture (Allen, Ferguson, & Stent, 2001)) because, in order for such comparisons to actually be useful rather than perfunctory, they require a developer's level of understanding of each environment (not available in publications and always in flux), extensive critical analysis, and presentation space for background about all of the environments under comparison. Readers interested in more broad-based literature reviews will find Jurafsky & Martin (2009), Clark, Fox, & Lappin (2010), and Jackendoff (2012) to be good starting points.

high-quality natural-language-to-metalanguage translations: (1) *Computational formal semantics*, like its non-computational counterpart, focuses on determining the truth conditions of declarative sentences, interpreting nondeclarative sentences based on what would make the declarative variant true, and interpreting quantifiers (Blackburn & Bos, 2005).[6] (2) *Automatic inferencing* seeks to enable systems to infer from an input like "The Mona Lisa hangs in Paris" that *The Mona Lisa is in France* (to take an example from Manning, 2006). (3) *Mindreading* using abductive reasoning permits systems to infer the mental states of others, which is a necessary capability for effective communication during joint activities (Langley et al., 2014).

Among the currently hot topics in the reasoning community is reasoning about narratives. For example, Finlayson (2015) reports a system that can learn plot functions – such as "Villany/Lack", "Struggle & Victory", and "Reward" – in folktales from semantically annotated versions of those texts. The annotations were recorded manually or semi-automatically since they covered many features that cannot be computed with high reliability given the current state of the art, such as the temporal ordering of events, mappings to WordNet senses, event valence (the event's impact on the Hero), and the identification of *dramatis personae* (i.e., character types). Providing those features automatically with the help of deep NLU is an obvious next step.

Another system addressing story understanding is Winston's (2012) *Genesis*. It carries out common-sense reasoning, including, for example, identifying that a concept like *revenge* plays a role in a story despite the absence of the word *revenge* or any of its synonyms. Genesis uses as input what Winston refers to as simple plot summaries written in English. Although the linguistic nature of the plot summaries would not be important to the casual reader, it is for this discussion of NLU. The reason is that the simplicity is not only linguistic: in terms of content and organization, the summaries look more like logical forms rendered in stylized English than what we would typically view as a plot summary. For example, the summary for Cyberwar begins: "Cyberwar: Estonia and Russia are countries. Computer networks are artifacts. Estonia insulted Russia because Estonia relocated a war memorial." This summary excerpt includes both unexpected definitional components (one would expect this information to be available in a story-independent knowledge base) and a non-canonical use of the closed-class item *because*. Our point is not that such inputs are inappropriate: they are useful and entirely fitting in support of research whose focus lies outside of NLP. However, it is important to recognize that moving from this style of input to more customary natural language plot summaries will require not only more robust NLU, but also more robust reasoning with respect to content and organization.

Turning to the robotics connection, the robotic systems reported by Lindes and Laird (2016) and Scheutz et al. (2017) incorporate natural language processing. The former system implements a parser based on embodied construction grammar (Feldman, Dodge, & Bryant, 2009), while the latter concentrates on translating a limited set of natural language utterances into a "Lambda calculus representation of words [that] could be inferred in an inverse manner from examples of sentences and their formal representation" (Baral, Lumpkin, & Scheutz, 2017, p. 11). In both systems, the role of the language component is to support a) direct human-robotic interaction, predominantly simple commands; and b) robotic learning of the meanings of words as the means

[6] It can also be used to determine the consistency of databases using theorem provers, but this angle is not necessarily heavily dependent upon NLU.

of grounding linguistic expressions in the robot's world model. As a result of the above choice, the extent of the coverage of language phenomena as well as the robot's conceptual knowledge is to support the needs of the robotic system. At present, these needs are limited on both counts. If the ultimate goal is to develop robotic language understanding that approaches human-level sophistication, then the above systems will have to tackle a plethora of NLU issues that their current objectives allow them to postpone.

10. Broader Issues

This paper sketched a multi-stage process of deep language understanding by LEIAs, with an emphasis on treating a large inventory of linguistic phenomena in an organized, psychologically-motivated way that fosters both near-term results and progress over the long term. The brief descriptions here are expanded upon in our book in preparation, *Language-Endowed Intelligent Agents* (M. McShane & S. Nirenburg). Our approach differs from mainstream NLP in many ways: it pursues deep, contextual semantic analysis; it integrates NLU within overall agent cognition and recognizes that only a holistic approach has any chance of ultimate success (i.e., splitting off NLU as a separate task is not a simplification – it actually makes the task completely impossible); it embraces the "scruffy" nature of actual language use; and it does not pursue the unnecessary goal of arriving at full and complete analyses of every input. Our approach also distinguishes domain-independent knowledge and reasoning from domain-specific knowledge and reasoning, thereby both clarifying and maximizing what can be reused across agents and applications.

The methodology of LEIA development involves proof-of-concept implementations under conditions that are sufficiently realistic to a) validate the component microtheories that treat individual phenomena and b) give us good reason to believe that the approach should scale up in sync with the growth of the knowledge resources. For example, although the lexicon at present contains only around 30,000 senses of words and phrases – not nearly enough to support NLU in all domains – it is sufficiently polysemous to require LEIAs to fundamentally treat lexical ambiguity. LEIAs, right now, are dealing with over 40 senses of the verb *make* (many of which are phrasals), over 20 senses of *have*, 18 for the preposition *in*, and so on. Similarly, our environment already fulfills all the prerequisites for taking on difficult problems like the interaction of lexical and referential ambiguity, unexpected input, learning new words on the fly, and detecting and reconstructing elided material. If our methods for treating such phenomena are shown to work for inputs covered by the current lexicon and ontology – which we have painstakingly prevented from containing the kinds of oversimplifications that would result in misleadingly impressive evaluations – then they should work equally well over larger versions of those resources.

It is impossible for us to make formal statements about the level of implementation of the system in this paper.[7] Formal evaluations exist to erase ambiguity from claims concerning results of experimentation; we have no formal evaluations to report about the integrated system, and, therefore, we are making no claims about its overall status. Over the years, we have reported

[7] We are commenting on implementation, scalability and evaluation only to respond to reviewer comments. The space is too short to adequately treat these complex issues.

formal evaluations of select system components (McShane, Nirenburg, & Beale, 2015, 2016; McShane, Beale, & Babkin, 2014). These were carried out using an earlier, non-incremental, implementation of our semantic analyzer. The findings that still remain valid are those that relate to conceptual, rather than numerical, results. For example, the evaluation of multiword expressions identified more frequent ambiguity in multiword expressions than we had anticipated. The constantly changing state of system implementations in a long-term research program raises important questions about the evaluation of knowledge-based systems overall.

For knowledge-based systems, the extent of coverage and analytical quality of component microtheories – including their heuristic content as well as their clarity, succinctness, integration with other microtheories, computational tractability, and so on – are as important, if not more important, than what happens to have been implemented in computer programs at a given point in time, particularly since implementations always lag behind algorithmic specifications. Over decades of working on NLU, we have found *in*formal evaluations to be a useful and necessary tool in vetting microtheories and ensuring their computational tractability. By contrast, we have found *formal* evaluations to be so resource-hungry that it is difficult to justify spending the necessary effort. Even more importantly, no formal evaluation of a knowledge-based system can, in fact, be carried out at this time without constraining the world, situation, and task so severely as to reduce them to a very marginally extended blocks world. This is an open secret to most members of our community, though most people behave as if it does not exist and still go through formal evaluations under severely unrealistic conditions, presumably to adhere to the prevalent methodological trends in the AI of today.

The resource-intensive nature of formal evaluations goes largely unrecognized in statistical NLP circles because evaluation guidelines and resources for many tasks are made available to the community at no cost to developers. This frees developers from having to: a) formulate task specifications, including rule-in/rule-out criteria; b) provide justifications for those specifications, which can be particularly thorny for the evaluation of individual functionalities rather than end systems; c) create evaluation suites; and d) provide gold standards against which precision and recall can be measured. In addition, the datasets accompanying task specifications often include perfectly computed upstream processing results (i.e., gold-standard manual annotations) in order to ensure that the target capability is evaluated "cleanly". If we impose on knowledge-based systems these same requirements for, on the one hand, formally detailing the evaluation set-up for every microtheory, and, on the other hand, making the evaluation of every capability "clean" by manually providing perfect results for all related processing, we will crush the enterprise in its infancy. Moreover, there is no guarantee that, even if this work were undertaken, the results would be judged by the community as appropriate or sufficient.

Let us take just one example to illustrate. Say we want to formally evaluate the microtheory of NN compounding (which we did, in fact, undertake in McShane, Beale, & Babkin, 2014). Analyzing an NN compound requires analyzing its semantic integration into the context, so lexical disambiguation and the establishment of the semantic dependency structure must all work correctly in the system that is being prepared to evaluate NN compounding capabilities. So, too, must coreference resolution, since disambiguating one of the Ns might require coreference with an earlier mention. And the new word learning module might need to be brought in as well, since one of the Ns might not be known. And then there's recovery from unexpected syntax, since the parse of a given sentence containing an NN might not work by fault of the parser – which means that this module must also be a part of the evaluation set-up, if the evaluation is to be trusted at all and not just be carried out under make-believe assumptions just to check the necessary box on the methodological checklist. So, how does one tease NN evaluation out of all of this? With great difficulty. In our 2014 experiment, we were able to evaluate only a subset of our NN analysis methods and were forced to make a number of unavoidable simplifications.

This brings us to another method of evaluating progress in NLU – evaluating an end application system that includes an NLU component. The assumption is that if such a system works well then so must its NLU component. This might be the gold standard for intelligent agents a couple of decades from now, but it will be only partially elucidating in the short term for two reasons. First, agent capabilities tend currently to be so limited (in terms of reasoning and action) that fully sufficient language support can easily be hacked, thus obviating the need to employ an extensive and deep understander of the kind we describe in this paper. This is, in fact, happening in the field and can be explained by the need to check the abovementioned methodological boxes. But this approach does not contribute to evaluating the quality of NLU. The second reason why evaluating NLU in end applications is not entirely satisfactory is that individuals – and, presumably, robot-human teams – can collaborate quite effectively in some domains with near zero language communication. An example that comes to mind involves our collaborator on the Maryland Virtual Patient system, Dr. Bruce Jarrell. Apart from being a surgeon and pedagogue, he is an artistic blacksmith who collaborated with a Ukrainian-speaking blacksmith on a remarkable window sculpture of an elm tree, despite having no language in common.[8] In short, system-level evaluation, like microtheory-level evaluation, has the potential to be useful, but only if the system requires that the agent engage in human levels of communication.

In closing, the evaluation of knowledge-based systems is an important, open methodological issue, one that our community must give more serious attention.

Acknowledgements

This research was supported in part by Grants N00014-16-1-2118 and N00014-17-1-2218 from the U.S. Office of Naval Research. Any opinions or findings expressed in this material are those of the authors and do not necessarily reflect the views of the Office of Naval Research.

References

Allen, J., Ferguson, G., & Stent, A. (2001). An architecture for more realistic conversational systems. *Proceedings of the Sixth International Conference on Intelligent User Interfaces* (pp. 1−8).

Baral, C., Lumpkin, B., & Scheutz, M. (2017). A high level language for human-robot interaction. *Poster Collection of the Fifth Annual Conference on Advances in Cognitive Systems*. Troy, NY: Cognitive Systems Foundation.

[8] For a photograph, see `https://archive.hshsl.umaryland.edu/bitstream/10713/2250/5/sculpturea.jpg`.

Blackburn, P., & Bos, J. (2005). *Representation and inference for natural language: A first course in computational semantics*. Stanford, CA: Center for the Study of Language and Information.

Clark, A., Fox, C., & Lappin, S. (Eds.). (2010). *The handbook of computational linguistics and natural language processing*. Chichester, United Kingdom: Wiley-Blackwell.

Cycorp publications. Available at `http://www.cyc.com/publications/`

Dzifcak, J., Scheutz, M., Baral, C., & Schermerhorn, P. (2009). What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management and action execution. *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*. Kobe, Japan: IEEE Press.

English, J., & Nirenburg, S. (2010). Striking a balance: Human and computer contributions to learning through semantic analysis. *Proceedings of the Fourth IEEE International Conference on Semantic Computing* (pp. 16–23). Pittsburgh, PA: IEEE Press.

Feldman, J., Dodge, E., & Bryant, J. (2009). Embodied construction grammar. In B. Heine & H. Narrog (Eds.), *The Oxford handbook of linguistic analysis*. New York: Oxford University Press.

Finlayson, M. A. (2015). Inferring Propp's functions from semantically-annotated text. *Journal of American Folklore*, *129*, 55−77.

Jackendoff, R. (2012). Language. In K. Frankish & W. M. Ramsey (Eds.), *The Cambridge handbook of cognitive science*, 171−192. Cambridge, England: Cambridge University Press.

Jurafsky, D., & Martin, J. H. (2009). *Speech and language processing: An introduction to natural language processing, speech recognition, and computational linguistics* (2nd Ed.). Upper Saddle-River, NJ: Prentice-Hall.

Langley, P., Meadows, B., Gabaldon, A., & Heald, R. (2014). Abductive understanding of dialogues about joint activities. *Interaction Studies*, *15*, 426−454.

Lindes, P., & Laird, J. E. (2016). Toward integrating cognitive linguistics and cognitive language processing. *Proceedings of the Fourteenth International Conference on Cognitive Modeling*. University Park, Pennsylvania: Penn State University Press.

Manning, C. D. (2006). *Local textual inference: It's hard to circumscribe, but you know it when you see it—and NLP needs it*. Unpublished manuscript accessed on July 28, 2018 from `https://nlp.stanford.edu/manning/papers/LocalTextualInference.pdf`.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60. Stroudsburg, PA: The Association for Computational Linguistics.

McShane, M. (2009). Reference resolution challenges for an intelligent agent: The need for knowledge. *IEEE Intelligent Systems*, *24*, 47–58.

McShane, M. (2015). Expectation-driven treatment of difficult referring expressions. *Proceedings of the Third Annual Conference on Advances in Cognitive Systems*. Atlanta, GA: Cognitive Systems Foundation.

McShane, M., & Babkin, P. (2016). Detection and resolution of verb phrase ellipsis. *Linguistic Issues in Language Technology, 13,* 1−34.

McShane, M., Beale, S., & Babkin, P. (2014). Nominal compound interpretation by intelligent agents. *Linguistic Issues in Language Technology*, *10*, 1–34.

McShane, M., Blissett, K., & Nirenburg, I. (2017). Treating unexpected input in incremental semantic analysis. *Proceedings of the Fifth Annual Conference on Advances in Cognitive Systems*. Troy, NY: Cognitive Systems Foundation.

McShane, M., & Nirenburg, S. (2012). A knowledge representation language for natural language processing, simulation and reasoning. *International Journal of Semantic Computing*, *6*, 3–23.

McShane, M., & Nirenburg, S. (In press). *Language-endowed intelligent agents*.

McShane, M., Nirenburg, S., & Beale, S. (2015). The Ontological Semantic treatment of multiword expressions. *Lingvisticæ Investigationes*, *38*, 73–110.

McShane, M., Nirenburg, S., & Beale, S. (2016). Language understanding with Ontological Semantics. *Advances in Cognitive Systems*, *4*, 35−55.

Nirenburg, S., & McShane, M. (2016a). Natural language processing. In S. E. F. Chipman (Ed.), *The Oxford handbook of cognitive science.* Volume 1. New York: Oxford University Press.

Nirenburg, S., & McShane, M. (2016b). Slashing metaphor with Occam's Razor. *Proceedings of the Fourth Annual Conference on Advances in Cognitive Systems.* Evanston, IL: Cognitive Systems Foundation.

Nirenburg, S., McShane, M., & Beale, S. (2008). A simulated physiological/cognitive "double agent". *Papers from the AAAI Fall Symposium on Biologically Inspired Cognitive Architectures* (pp. 127−134). Menlo Park, CA: AAAI Press.

Nirenburg, S., McShane, Beale, S., M., Wood, P., Scassellati, B., Mangin, O., & Roncone, A. (2018). Toward human-like robot learning. *Proceedings of the Twenty-Third International Conference on Natural Language and Information Systems* (pp. 73−82). Paris, France: Springer.

Nirenburg, S., & Raskin, V. (2004). *Ontological semantics*. Cambridge, MA: The MIT Press.

Scheutz, M., Krause, E., Oosterveld, B., Frasca, T., & Platt, R. (2017). Spoken instruction-based one-shot object and action learning in a cognitive robotic architecture. *Proceedings of the Sixteenth International Conference on Autonomous Agents and Multiagent Systems* (pp. 1378–1386). São Paulo, Brazil: ACM.

Winston, P. (2012). The right way. *Advances in Cognitive Systems*, *1*, 23−36.