Linguistics for the Age of AI

The title-level DOI for this work is:

doi:10.7551/mitpress/13618.001.0001

1

Our Vision of Linguistics for the Age of AI

1.1 What Is Linguistics for the Age of AI?

A long-standing goal of artificial intelligence has been to build intelligent agents that can function with the linguistic dexterity of people, which involves such diverse capabilities as participating in a fluent conversation, using dialog to support task-oriented collaboration, and engaging in lifelong learning through processing speech and text. There has been much debate about whether this goal is, in principle, achievable since its component problems are arguably more complex than those involved in space exploration or mapping the human genome. In fact, enabling machines to emulate human-level language proficiency is well understood to be an AI-complete problem—one whose full solution requires solving the problem of artificial intelligence in general. However, we believe that it is in the interests of both scientific progress and technological innovation to assume that this goal *is* achievable until proven otherwise. The question then becomes, How best to pursue it?

We think that a promising path forward is to pursue linguistic work that adheres to the following tenets:

- Language processing is modeled from the agent perspective, as one component of an integrated model of perception, reasoning, and action.
- The core prerequisites for success are the abilities to (a) extract the meaning of linguistic expressions, (b) represent and remember them in a model of memory, and (c) use these representations to support an intelligent agent's decision-making about action—be it verbal, physical, or mental.
- While extralinguistic information is required for extracting the full meanings of linguistic inputs, in many cases, purely linguistic knowledge is sufficient to compute an interpretation that can support reasoning about action.
- Language modeling must cover and integrate the treatment of all linguistic phenomena (e.g., lexical ambiguity, modality, reference) and all components of processing (e.g., syntax, semantics, discourse).

- The treatments of language phenomena are guided by computer-tractable micro-theories describing specific phenomena and tasks.
- A core capability is lifelong learning—that is, the agent's ability to independently learn new words, ontological concepts, properties of concepts, and domain scripts through reading, being told, and experience.
- Methodologically, the accent is on developing algorithms that facilitate the treatment of the many tasks within this research program. Any methods can be brought to bear as long as they are sufficiently transparent to allow the system's decisions to be explained in a manner that is natural for humans.

We call this program of work Linguistics for the Age of AI. The first half of this chapter describes the program in broad strokes. The deep dives in the second half provide additional details that, we think, might go beyond the interests of some readers.

1.2 What Is So Hard about Language?

For the uninitiated, the complexities of natural language are not self-evident: after all, people seem to process language effortlessly. But the fact that human language abilities are often taken for granted does not make them any less spectacular. When analyzed, the complexity of the human language facility is, in fact, staggering—which makes modeling it in silico a very difficult task indeed.

What, exactly, makes language hard for an artificial intelligent agent? We will illustrate the complexity using the example of ambiguity. Ambiguity refers to the possibility of interpreting a linguistic unit in different ways, and it is ubiquitous in natural languages. In order to arrive at the speaker's intended meaning, the interlocutor must select the contextually appropriate interpretation of the ambiguous entity. There are many types of ambiguity in natural language. Consider a few examples:

1. **Morphological ambiguity.** The Swedish word *frukosten* can have five interpretations, depending on how its component morphemes are interpreted. In the analyses below, lexical morphemes are separated by an underscore, whereas the grammatical morpheme for the definite article (*the*) is indicated by a plus sign:

   - frukost+en "the breakfast"
   - frukost_en "breakfast juniper"
   - fru_kost_en "wife nutrition juniper"
   - fru_kost+en "the wife nutrition"
   - fru_ko_sten "wife cow stone" (Karlsson, 1995, p. 28)

2. **Lexical ambiguity.** The sentence *I made her duck* can have at least the following meanings, depending on how one interprets the words individually and in combination: "I forced her to bend down," "I prepared food out of duck meat for her," "I prepared

food out of the meat of a duck that was somehow associated with her" (it might have belonged to her, been purchased by her, been raised by her), and "I made a representation of a duck that is somehow associated with her" (maybe she owns it, is holding it).

3. **Syntactic ambiguity.** In the sentence *Elaine poked the kid with the stick*, did Elaine poke the kid using a stick, or did she poke (using her finger) a kid who was in possession of a stick?
4. **Semantic dependency ambiguity.** The sentence *Billy knocked over the vase* is underspecified with respect to Billy's semantic role: if he did it on purpose, he is the agent; if not, he is the instrument.
5. **Referential ambiguity.** In the sentence *The soldiers shot at the women and I saw some of them fall*, who fell—soldiers or women?
6. **Scope ambiguity.** Does *big rivers and lakes* describe big rivers and big lakes or big rivers and lakes of any size?
7. **Pragmatic ambiguity.** When a speaker says, *I need help fixing the toaster*, is this asserting a fact or asking the interlocutor for help?

If these examples served as input to a machine translation system, the system would, in most cases, have to settle on a single interpretation because different interpretations would be translated differently. (The fact that ambiguities can sometimes be successfully carried across languages cannot be relied on in the general case.) While the need to select a single interpretation should be self-evident for some of the examples, it might be less clear for others, so consider some scenarios. Regarding (3), Russian translates the instrumental and accompaniment meanings of *with* in different ways, so that this ambiguity must be resolved explicitly. Regarding (5), in Hebrew, the third-person plural pronoun—which is needed to translate *them*—has different forms for different genders, so a translation system would need to identify either *the women* or *the soldiers* as the coreferent. Regarding (6), the ambiguity can be carried through to a language with the same adjective-noun ordering in noun phrases, but possibly not to a language in which adjectives follow their nouns. Regarding (7), although some language pairs may allow for speech act ambiguity to be carried through in translation, this escape hatch will be unavailable if the application involves a personal robotic assistant who needs to understand what you want of it.

The obvious response to the question of how to arrive at a particular interpretation is, *Use the context!* After all, people use the context effortlessly. But what does using the context actually mean? What is the context? How do we detect, categorize, and select the salient bits of context and then use them in understanding language? At the risk of some overgeneralization, we can say that the historical and contemporary scope of natural language processing (NLP) research reflects a wide variety of responses to these questions. At one extreme of the range of solutions—the so-called *knowledge-lean* approaches—the context is defined as a certain number of words appearing to the right and to the left of the

word whose interpretation is sought. So, the context is words, period. At the other extreme—the *knowledge-based* approaches—the context is viewed as the combination of a large number of features about language, the situation, and the world that derive from different sources and are computed in different ways. Leveraging more elements from the context improves the accuracy of language interpretation; however, this ability comes at a steep price. One of the purposes of this book is to demonstrate that intelligent agents can often derive useful interpretations of language inputs without having to invoke every aspect of knowledge and reasoning that a person would bring to bear. An agent's interpretations may be incomplete or vague but still be sufficient to support the agent's reasoning about action—that is, the interpretations are *actionable*. Orienting around actionability rather than perfection is the key to making a long-term program of work toward human-level natural language processing at once scientifically productive and practically feasible.

A terminological note: In its broadest sense, the term *natural language processing* refers to any work involving the computational processing of natural language. However, over the past few decades, NLP has taken on the strong default connotation of involving knowledge-lean (essentially, semantics-free) machine learning over big data. Therefore, in the historically recent and current context, there is a juxtaposition between NLP and what we are pursuing in this book: NLU, or natural language *understanding* (see the deep dive in section 1.6.3 for discussion). However, earlier in the history of the field, the term *natural language processing* did encompass all methods of automating the processing of natural language. The historical discussion below inevitably uses both the broad and the narrow senses of the term. The context should make clear which sense is intended in each case. Lucky for us our readers are human.

1.3 Relevant Aspects of the History of Natural Language Processing

Natural language processing was born as machine translation, which developed into a high-profile scientific and technological area already in the late 1940s.[1] Within a decade of its inception, machine translation had given rise to the theoretical discipline of computational linguistics and, soon thereafter, to its applied facets that were later designated as natural language processing (NLP). The eponymous archival periodical of the field, *Computational Linguistics*, started its existence in 1954 as *Mechanical Translation* and, in 1965–1970, was published as *Mechanical Translation and Computational Linguistics*. A perusal of the journal's contents from 1954 to 1970 (http://www.machine translation-archive .info/MechTrans-TOC.htm) reveals a gradual shift from machine translation–specific to general computational-linguistic topics. The original machine translation initiative has also influenced other fields of study, most importantly theoretical linguistics and artificial intelligence.

From the outset, machine translation was concerned with building practical systems using whatever method looked most promising. It is telling that the first programmatic

statement about machine translation, Warren Weaver's (1955 [1949]) famous memorandum, already suggests a few potential approaches to machine translation that can be seen as seeds of future computational-linguistic and NLP paradigms. Then, as now, such suggestions were influenced by the scientific and technological advances that captured the spirit of the times. Today, this may be big data and deep learning. Back then, Weaver was inspired by (a) results in early cybernetics, specifically McCulloch's artificial neurons (McCulloch & Pitts, 1943) and their use in implementing logical reasoning; (b) recent advances in formal logic; and (c) the remarkable successes of cryptography during World War II, which contributed to the development of information theory, on which Weaver collaborated with Shannon (Shannon & Weaver, 1964 [1949]). Inspiration from cybernetics can be seen as the seed of the connectionist approach to modeling language and cognition. The formal logic of Tarski, Carnap, and others underwent spectacular development and contributed to formal studies of the syntax and semantics of language as well as to the development of NLP systems. Shannon's information theory is the precursor of the currently ascendant statistical, machine learning–oriented approaches to language processing.

In machine translation research, it was understood early on that simplistic, word-for-word translation could not succeed and that understanding and rendering meaning were essential. It was equally understood that people disambiguate language in context. It is not surprising, therefore, that Weaver (1955 [1949]) suggests involving contextual clues in text analysis:

> If one examines the words in a book, one at a time through an opaque mask with a hole in it one word wide, then it is obviously impossible to determine, one at a time, the meaning of words. "Fast" may mean "rapid"; or it may mean "motionless"; and there is no way of telling which. But, if one lengthens the slit in the opaque mask, until one can see not only the central word in question but also say N words on either side, then, if N is large enough one can unambiguously decide the meaning.

The context-as-text-window method of analyzing text was in sync with the then-ascendant linguistic theory, structuralism, which did not accept unobservables in its repertoire (see the deep dive in section 1.6.1).[2] In view of this, it is not surprising that Weaver did not suggest a method of determining text meaning. The prevailing opinion was that the computational processing of meaning was not possible—this was the reason why Norbert Wiener, a pioneer of cybernetics, refused to join the early machine translation bandwagon and also why Yehoshua Bar Hillel, in the conclusion of his 1960 survey of a decade of machine translation research,[3] insisted that fully automatic, high-quality machine translation could not be an immediate objective of the field before much more work on computational semantics had been carried out. It is noteworthy that neither Wiener nor Bar Hillel believed that fully automatic, high-quality machine translation—and, by extension, high-quality NLP—could succeed without the treatment of meaning.

At the same time as semantics was failing to attract research interest, syntax was taking off in both the theoretical and computational realms. In theoretical work, the newly

ascendant school of mentalist theoretical linguists—the generative grammarians—isolated syntax from other cognitive and language-processing capabilities with the goal of explaining the hypothesized Universal Grammar. In NLP, for its part, the flagship research direction for decades was developing syntactic parsers based on ever more sophisticated formal grammar approaches, such as lexical functional grammar, generalized phrase structure grammar, and head-driven phrase structure grammar.[4]

The study of meaning was not, however, completely abandoned: philosophers and logicians continued to pursue it with an accent on its formal representation and truth-conditional semantics. Formal representations, and their associated formal languages, were needed because it was assumed that formal reasoning could only be carried out over formal representations of the meanings of propositions—not the messy (ambiguous, elliptical, fragmentary) strings of natural language. Truth-conditional semantics, for its part, was a cornerstone of work on artificial reasoners in AI.

It is not surprising that a program of work headed by philosophers and logicians (not linguists) did not concentrate on translating from natural language into the artificial metalanguage of choice, even though that was a prerequisite for automatic reasoning. In fact, automating that translation process remains, to this day, an outstanding prerequisite to incorporating machine reasoning into end systems that involve natural language.

The distinction between these two lines of work—translating from natural language into a formal language, and reasoning over that formal language—was recognized early on by Bar Hillel, making his observation of long ago as relevant now as it was then:

> The evaluation of arguments presented in a natural language should have been one of the major worries … of logic since its beginnings. However, … the actual development of formal logic took a different course. It seems that … the almost general attitude of all formal logicians was to regard such an evaluation process as a two-stage affair. In the first stage, the original language formulation had to be rephrased, without loss, in a normalized idiom, while in the second stage these normalized formulations would be put through the grindstone of the formal logic evaluator. … Without substantial progress in the first stage even the incredible progress made by mathematical logic in our time will not help us much in solving our total problem. (Bar Hillel, 1970, pp. 202–203)

From the earliest days of configuring reasoning-oriented AI applications, contributing NLP systems did benefit from one simplifying factor: a given NLP system needed to interpret only those aspects of text, meaning that its target reasoning engine could digest (rather than aim for a comprehensive interpretation of natural language semantics). Still, preparing NLP systems to support automatic reasoning was far from simple. A number of efforts were devoted to extracting and representing facets of text meaning, notably, those of Winograd (1972), Schank (1972), Schank and Abelson (1977), Wilks (1975), and Woods (1975). These efforts focused on the resolution of ambiguity, which required knowledge of the

context. The context, in turn, was understood to include the textual context, knowledge about the world, and knowledge about the speech situation.

Such knowledge needed to be formulated in machine-tractable form. It included, nonexhaustively, grammar formalisms specifically developed to support parsing and text generation, actual grammars developed within these formalisms, dictionaries geared toward supporting automatic lexical disambiguation, rule sets for determining nonpropositional (pragmatic and discourse-oriented) meaning, and world models to support the reasoning involved in interpreting propositions. Broad-scope knowledge acquisition of this complexity (whether for language-oriented work or general AI) was unattainable given the relatively limited resources devoted to it. The public perception of the futility of any attempt to overcome this so-called knowledge bottleneck profoundly affected the path of development of AI in general and NLP in particular, moving the field away from rationalist, knowledge-based approaches and contributing to the emergence of the empiricist, knowledge-lean, paradigm of research and development in NLP.

The shift from the knowledge-based to the knowledge-lean paradigm gathered momentum in the early 1990s. NLP practitioners considered three choices:

1. Avoid the need to address the knowledge bottleneck either by pursuing components of applications instead of full applications or by selecting methods and applications that do not rely on extensive amounts of knowledge.
2. Seek ways of bypassing the bottleneck by researching methods that rely on direct textual evidence, not ontologically interpreted, stored knowledge.
3. Address the bottleneck head-on but concentrate on learning the knowledge automatically from textual resources, with the eventual goal of using it in NLP applications.[5]

It is undeniable that the center of gravity in NLP research has shifted almost entirely toward the empiricist, knowledge-lean paradigm. This shift has included the practice of looking for tasks and applications that can be made to succeed using knowledge-lean methods and redefining what is considered an acceptable result—in the spirit of Church and Hovy's (1993) "Good Applications for Crummy Machine Translation." The empiricist paradigm was, in fact, already suggested and experimented with in the 1950s and 1960s—for example, by King (1956) with respect to machine translation. However, it became practical only with the remarkable advances in computer storage and processing starting in the 1990s.

The reason why NLP is particularly subject to fluctuations of fashion and competing practical and theoretical approaches is that, unlike other large-scale scientific efforts, such as mapping the human genome, NLP cannot be circumscribed by a unifying goal, path, purview, or time frame. Practitioners' *goals* range from incrementally improving search engines, to generating good-quality machine translations, to endowing embodied intelligent agents with language skills rivaling those of a human. *Paths* of development range from manipulating surface-level strings (words, sentences) using statistical methods, to

generating full-blown semantic interpretations able to support sophisticated reasoning by intelligent agents. The *purview* of an NLP-oriented R&D effort can range from whittling away at a single linguistic problem (e.g., how quantification is expressed in Icelandic), to developing theories of selected language-oriented subdisciplines (e.g., syntax), to building full-scale, computational language understanding and/or generation systems. Finally, the *time frame* for projects can range from months (e.g., developing a system for a competition on named-entity recognition) to decades and beyond. Practically the only thing that NLP practitioners do agree on is just how difficult it is to develop computer programs that usefully manipulate natural language—a medium that people master with such ease.

Kenneth Church (2011) presents a compelling analysis of 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 embrace 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). (p. 3)

In this must-read 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 also for practical ones:

> 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. (p. 7)

Dovetailing with Church, we have identified a number of opinion statements—detailed in the deep dive in section 1.6.3—that have led to a puzzling putative competition between knowledge-lean and knowledge-based approaches, even though they are pursuing entirely different angles of AI.

1.4 The Four Pillars of Linguistics for the Age of AI

The above perspective on the state of affairs in the field motivates us to define Linguistics for the Age of AI as a distinct perspective on the purview and methods of linguistic work.

This perspective rests on the following four pillars, which reflect the dual nature of AI as science and practice.

1. Language processing capabilities are developed within an integrated, comprehensive agent architecture.
2. Modeling is human inspired in service of explanatory AI and actionability.
3. Insights are gleaned from linguistic scholarship and, in turn, contribute to that scholarship.
4. All available heuristic evidence is incorporated when extracting and representing the meaning of language inputs.

We now consider each of these in turn.

1.4.1 Pillar 1: Language Processing Capabilities Are Developed within an Integrated, Comprehensive Agent Architecture

Since at least the times of Descartes, the scientific method has become more or less synonymous with the analytic approach, whereby a phenomenon or process is decomposed into contributing facets or components. The general idea is that, after each such component has been sufficiently studied independently, there would follow a synthesis step that would result in a comprehensive explanation of the phenomenon or process. A well-known example of the application of the analytic approach is the tenet of the autonomy of syntax in theoretical linguistics, which has been widely adopted by—and has strongly influenced—the field of computational linguistics. The analytic approach makes good sense because it is well-nigh impossible to expect to account for all the facets of a complex phenomenon simultaneously and at a consistent grain size of description. But it comes with a cost: it artificially constrains the purview of theories and the scope of models, and often unwittingly fosters indefinite postponement of the all-important synthesis step.

If we step back to consider some of the core tasks of a language-enabled intelligent agent, we see how tightly integrated they actually are and why modularization is unlikely to yield results if not complemented by the concern for integration. Which functionalities will have to be integrated? As a first approximation,

- Agents must implement some version of a BDI (belief-desire-intention) approach to agent modeling (Bratman, 1987) to make manifest how they select plans and actions to fulfill their goals.
- They must learn, correct, and augment their knowledge of the world (including their knowledge about themselves and other agents), as well as their knowledge of language, through experience, reasoning, reading, and being told.
- They must communicate with people and other agents in natural language.

- They must model experiencing, interpreting, and remembering their own mental, physical, and emotional states.
- They must manage their memories—including forgetting and consolidating memories.
- They must model and reason about the mental states, goals, preferences, and plans of self and others, and use this capability to support collaboration with humans and other intelligent agents.
- And, if they are embodied, they will require additional perception modalities, support for physical action, and, at least in a subset of applications, a simulated model of human physiology.

In order to minimize development effort, maximize resource reuse, and avoid knowledge incompatibilities, it is preferable to support all these processes within an integrated knowledge substrate encoded in an interoperable knowledge representation language.[6] Note that this requirement is not motivated theoretically; it is purely ergonomic, since significant engineering is needed to integrate different formalisms and approaches to knowledge representation in a single system.

The OntoAgent cognitive architecture referenced throughout the book has been designed with the above suite of functionalities in mind. Figure 1.1 shows a high-level (and ruthlessly simplified) view of that architecture, which will be refined in future chapters (see especially figure 7.1) to the degree necessary for explaining the linguistic behavior of language-endowed intelligent agents (LEIAs).

Agents obtain new facts about the world both through analyzing sensory inputs and as a result of their own mental actions. Attention to these new facts may trigger the adding of goals to the goal agenda. At each operation cycle, the agent prioritizes the goals on the agenda and then selects the plan(s) that will result in some physical, verbal, or mental action(s).

The core knowledge resources of the architecture include an ontological model (long-term semantic memory), a long-term episodic memory of past conscious experiences, and a situation model that describes the participants, props, and recent events in the current situation. The ontological model includes not only general world knowledge but also an inventory of the agent's goals; its physical, mental, and emotional states; its long-term personality traits and personal biases; societal rules of behavior, including such things as knowledge about the responsibilities of each member of a task-oriented team; and the agent's model of the relevant subset of the abovementioned features of its human and agentive collaborators.[7] The situation model, for its part, includes not only the representation of a slice of the observable world but also the agent's beliefs about its own and other agents' currently active goal and plan instances, as well as their current physical, mental, and emotional states.

The agent's knowledge enables its conscious decision-making as well as its ability to explain its decisions and actions. For the purposes of this book, the important point is that

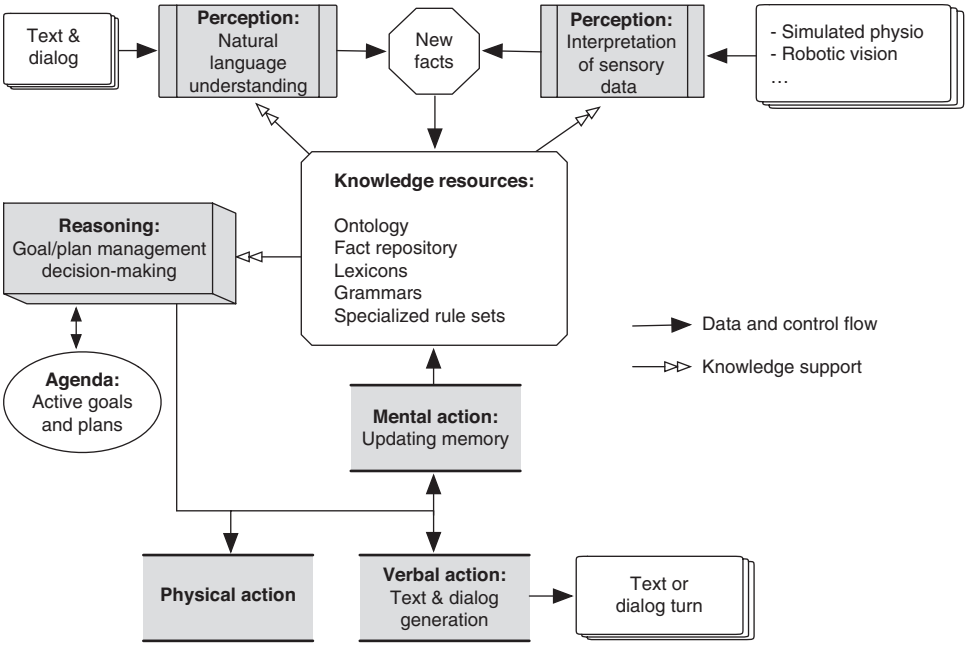

Figure 1.1
High-level sketch of the OntoAgent architecture.

this knowledge—both static and dynamically computed—is necessary for deriving the full meaning of language inputs. The view of agency we are sketching here is broadly similar to well-known approaches in cognitive modeling and AI, for example, the general worldview of such cognitive architectures as SOAR (Rosenbloom et al., 1991) or the BDI movement (Bratman, 1987).

In this book, we concentrate on those capabilities of LEIAs that are germane to their language understanding functionalities. When the LEIA receives text or dialog input (upper left in figure 1.1), it interprets it using its knowledge resources and a battery of reasoning engines, represented by the module labeled *Perception: Natural language understanding.* The internal organization and the functioning of this module is what this book is about. The result of this module's operation is one or more *New facts*, which are unambiguous assertions written in the metalanguage shared across all of the agent's knowledge resources and all downstream processing modules. These facts are then incorporated into the agent's knowledge bases.

As can be seen in figure 1.1, *New facts* can be obtained through channels of perception other than language. Robotic vision, other sensors, and even computer simulations (e.g., of human physiology) can all serve as sources of new information for a LEIA. And just like language, they must be interpreted using the module marked *Perception: Interpretation of*

sensory data in the figure. This interpretation results in the same kinds of *New facts*, written in the same metalanguage, as does language understanding.[8] The upshot is that all knowledge learned by the agent from any source is equally available for the agent's subsequent reasoning and decision-making about action. The generation of associated actions—which can be physical, mental, or verbal—also involves extensive knowledge and reasoning since the actions must be selected and planned before actually being carried out. In this book, we will concentrate on a detailed and comprehensive exploration of how LEIAs understand language, while not detailing the processes through which certain components of the agent's internalized knowledge were obtained as a result of perception other than language understanding.

The above sketch of the OntoAgent architecture is high-level and omits a wealth of specialist detail. We include it here simply to frame the process of language understanding that constitutes the core of Linguistics for the Age of AI. Additional details about OntoAgent will be provided throughout this book whenever required to clarify a particular facet of language processing.

The OntoAgent approach to language processing is methodologically compatible with the cognitive systems paradigm in that it focuses on natural *understanding* in contrast to semantically impoverished natural language *processing* (Langley et al., 2009). Other language understanding efforts within this paradigm (e.g., Lindes & Laird, 2016; Cantrell et al., 2011)—while not sharing all the same assumptions or pursuing the same depth and breadth of coverage as OntoAgent—are united in that they all pursue the goal of faithfully replicating human language understanding behavior as a part of overall humanlike cognitive behavior. (For more on cognitive systems overall, see the deep dive in section 1.6.4.) The extent, quality, and depth of language understanding in each individual approach is determined by the scope of functionalities of the given cognitive agent—not independently, as when natural language processing is viewed as a freestanding task. Consequently, these approaches must take into account nonlinguistic factors in decision-making, such as the long-term and short-term beliefs of the given agent, its biases and goals, and similar features of other agents in the system's environment.

Consider, for example, anticipatory text understanding, in which an agent can choose to act before achieving a complete analysis of a message, and possibly before even waiting for the whole message to come through—being influenced to do so, for example, by the principle of economy of effort. Of course, this strategy might occasionally lead to errors, but it is undeniable that people routinely pursue anticipatory behavior, making the calibration of the degree of the anticipation an interesting technical task for cognitively inspired language understanding. Anticipatory understanding extends the well-known phenomenon of priming (e.g., Tulving & Schacter, 1990) by relying on a broader set of decision parameters, such as the availability of up-to-date values of situation parameters, beliefs about the goals and biases of the speaker/writer, and general, ontological knowledge about the world.

Our emphasis on comprehensive cognitive modeling naturally leads to a preference for multilayered models. We distinguish three levels of models, from the most general to the most specialized:

1. The *cognitive architecture* accounts for perception, reasoning, and action in a tightly integrated fashion.
2. The *NLU module* integrates the treatment of a very large number of linguistic phenomena in an analysis process that, we hypothesize, emulates how humans understand language.
3. The specialized models within the NLU module, called *microtheories*, treat individual linguistic phenomena. They anticipate and seek to cover the broadest possible scope of manifestations of those phenomena.

This infrastructure facilitates the exploration and development of detailed solutions to individual and interdependent problems over time. An important feature of our overall approach is that we concentrate not only on architectural issues but also, centrally, on the heuristics needed to compute meaning.

We have just explained the first part of our answer to the question, *What is Linguistics for the Age of AI?* It is the study of linguistics in service of developing natural language understanding and generation capabilities *within an integrated, comprehensive agent architecture*.

1.4.2 Pillar 2: Modeling Is Human Inspired in Service of Explanatory AI and Actionability

In modeling LEIAs, we are not attempting to replicate the human brain as a biological entity. Even if that were possible, it would fail to serve one of our main goals: *explanatory power*. We seek to develop agents whose behavior is explainable in human terms by the agents themselves. As an introductory example of the kinds of behavior we address in our modeling, consider the following situation. Lavinia and Felix are in an office with an open window in late fall. Lavinia says, "It's cold in here, isn't it?" Felix may respond in a variety of ways, including the following:

1. Yes, it *is* rather cold.
2. Do you want me to close the window?

Response (1) demonstrates that Felix interpreted Lavinia's utterance as a question and responded affirmatively to it. Response (2) demonstrates that Felix

a. interpreted Lavinia's utterance as an indirect request;
b. judged that Lavinia had an appropriate social status to issue this request;
c. chose to comply;

d. selected the goal of making the room warmer (rather than, say, making Lavinia warmer—as by offering a sweater);

e. selected one of the plans he knew for attaining this goal; and

f. decided to verify that carrying out this plan was preferable to Lavinia before doing it.

We want our agents to not only behave like this but also be able to explain why they responded the way they did in ways similar to (a)–(f). In other words, our models are inspired by our folk-psychological understanding of how people interpret language, make decisions, and learn. The importance of explainable AI cannot be overstated: society at large is unlikely to cede important decision-making in domains like health care or finance to machines that cannot explain their advice. For more on explanation, see the deep dive in section 1.6.5.

Our model of NLU does not require that agents exhaustively interpret every input to an externally imposed standard of perfection. Even people don't do that. Instead, agents operating in human-agent teams need to understand inputs to the degree required to determine which goals, plans, and actions they should pursue as a result of NLU. This will never involve blocking the computation of a human-level analysis if that is readily achievable; it will, however, absolve agents from doggedly pursuing ever deeper analyses if it is unnecessary in a particular situation.

In other words, in our models, agents decide how deeply they need to understand an input, and what counts as a successful—specifically, *actionable*—interpretation, based on their plans, goals, and overall understanding of the situation. If the goal is to learn new facts, then complete understanding of the portion of text containing the new fact might be preferable. By contrast, if an agent hears the input *We are on fire! Grab the axe. We need to hack our way out!*, it should already be moving toward the axe before working on interpreting the final sentence. In fact, a meaning representation that is sufficient to trigger an appropriate action by an agent may even be vague or contain residual ambiguities and lacunae.

Actionability-oriented human behavior can be explained in terms of the principle of least effort (Zipf, 1949). Piantadosi et al. (2012) argue that maintaining a joint minimum of effort between participants in a dialog is a universal maximizing factor for efficiency in conversation. Speakers do not want to spend excessive effort on precisely specifying their meaning; but hearers, for their part, do not want to have to apply excessive reasoning to understand the speaker's meaning. A core prerequisite for minimizing effort in communication is for the dialog participants to have models of the other's beliefs, goal and plan inventories, personality traits, and biases that allow them to "mindread" each other and thus select the most appropriate amount of information to convey explicitly. People use this capability habitually. It is thanks to our ability to mindread that we will describe a bassoon as "a low double reed" only in conversation with musicians. (For more on mindreading, see chapter 8.)

Another vestige of the operation of the principle of least effort in our work is our decision to have our agents look for opportunities to avoid having to resolve all ambiguities in a given input, either postponing this process (allowing for underspecification) or

pronouncing it unnecessary (recognizing benign ambiguity). This is a core direction of our research at the intersection of NLP and cognitive science.

Having worked for years on developing an agent system to teach clinical medicine, we see a compelling analogy between building LEIAs and training physicians. Clinical medicine is a notoriously difficult domain: the volume of research is growing at an unprecedented rate, but the scientific knowledge that can be distilled from it is still inadequate to confidently answer all clinical questions. As a result, the field is arguably still as much art as science (Panda, 2006). And yet medical schools produce competent physicians. These physicians have different mental models of medicine, none of which is complete or optimal—and yet, they practice and save lives. Developers of AI systems need to adopt the same mindset: a willingness to take on the problem of human cognition—which is, in a very real sense, *too hard*—and make progress that will serve both science and society at large.

This concludes the explanation of the second part of our answer to the question, *What is Linguistics for the Age of AI?* It is the study of linguistics in service of developing natural language understanding and generation capabilities (1) within an integrated, comprehensive agent architecture, (2) *using human-inspired, explanatory modeling techniques and actionability judgments*.

1.4.3 Pillar 3: Insights Are Gleaned from Linguistic Scholarship and, in Turn, Contribute to That Scholarship

The past seventy years have produced a tremendous amount of scholarship in linguistics and related fields. This includes theories, data analyses, print and digital knowledge bases, corpora of written and spoken language, and experimental studies with human subjects. It would be optimal if all these fruits of human thinking could somehow converge into artificial intelligence; but, alas, this will not happen. In fact, not only will there be no smooth convergence, but much of the scholarship is not applicable to the goals and requirements of AI for the foreseeable future. While this is a sobering statement, it is not a pessimistic one: it simply acknowledges that there is a fundamental difference between human minds as thoughtful, creative consumers of scholarship and machines as nonthinking, exacting demanders of algorithms (despite the overstretched metaphorical language of *neural networks* and *machine learning*). Stated differently, it is important to appreciate that much of linguistic scholarship involves either theoretical debates that float above a threshold of practical applicability or human-oriented descriptions that do not lend themselves to being formulated as computable heuristics.

Work in computational linguistics over the past twenty years or so has largely concentrated on corpus annotation in service of supervised machine learning.[9] During this time, the rest of the linguistics community has continued to work separately on human-oriented research. This has been an unfortunate state of affairs for developing LEIAs because neither the computational, nor the theoretical, nor the descriptive linguistic community has

been developing explanatory, heuristic-supported models of human language understanding that are directly suitable for implementation in agent systems.

By contrast, the models we seek to build, which we call *microtheories,* are machine-tractable descriptions of language phenomena that guide the agent, in very specific ways, through the language analysis process. Although microtheory development can be informed by noncomputational approaches, the main body of work in building a microtheory involves (a) determining the aspects of linguistic descriptions that are, in principle, machine-tractable and (b) developing the heuristic algorithms and knowledge needed to operationalize those descriptions. To take a simple example, lexicographers can explain what the English word *respectively* means, but preparing a LEIA to semantically analyze sentences that include *respectively*—for example, *Our dog and our cat like bones and catnip, respectively*—requires a dynamic function that effectively recasts the input as *Our dog likes bones and our cat likes catnip* and then semantically analyzes those propositions.

It would be a boon to agent development if linguists working in noncomputational realms would join the computational ranks *as well*. Such crossover-linguists would identify aspects of their theories and models that can be accounted for using precise, computer-tractable heuristics and then formulate the associated algorithms and descriptions. This work would not only serve NLU but, in all likelihood, also shed light on the theories and models themselves since the demands of computation set the bar of descriptive adequacy very high. In this section, we briefly review some sources of past inspiration from various fields as a prelude to what we hope will be a much richer mode of interaction in the future.[10]

1.4.3.1 Theoretical syntax Theoretical approaches to syntax attempt to account for the nature of the human language faculty with respect to sentence structure. Under this umbrella are approaches that range from almost exclusively theoretical to a combination of theoretical and descriptive. Some focus exclusively on syntax, whereas others consider interactions with other modules, such as semantics.

An example of a squarely theoretical, almost exclusively syntactic, approach is generative grammar in the tradition of Noam Chomsky. In its more recent manifestations (Chomsky, 1995), it is too abstract, too modular, and too quickly changing to inform practical system building. However, Chomsky's early work in this paradigm (e.g., Chomsky, 1957) spurred the development of the context-free grammars and associated parsing technologies that have been a cornerstone of natural language processing for decades.

Turning to theoretical approaches with practical applicability, a good example is construction grammar in its various manifestations (Hoffman & Trousdale, 2013). Construction grammars focus on the form-to-meaning mappings of linguistic entities at many levels of complexity, from words to multiword expressions to abstract templates of syntactic constituents. As theoretical constructs, construction grammars make particular claims about how syntactic knowledge is learned and organized in the human mind. For example, constructions are defined as learned pairings of form and function, their meaning is associated

exclusively with surface forms (i.e., there are no transformations or empty categories), and they are organized into an inheritance network. For agent modeling, what is most important is not the theoretical details (e.g., the role of inheritance networks) but (a) the basic insight—that is, that constructions are central to human knowledge of language—and (b) the descriptive work on the actual inventory and meaning of constructions.[11]

Our third example of a theoretical-syntax approach that can inform agent modeling is Dynamic Syntax (Kempson et al., 2001). It places emphasis on the incremental generation of decorated tree structures that are intended to capture not only the syntactic structure but also the semantic interpretation of utterances. Like the other theories mentioned here, this is a theory of language processing in humans, not by machines.[12] However, it reflects a core capability of human language processing that must be emulated if machines are to behave like humans: incremental, integrated syntactic and semantic analysis.

1.4.3.2 Psycholinguistics As we just saw, incrementality has been folded into the study of theoretical syntax, but it has also been a focus of investigation in the field of psycholinguistics. Experiments have established that language processing integrates linguistic and nonlinguistic sources of information as people understand inputs incrementally. For example, Altmann and Kamide (1999) report an experiment in which participants were shown a scene containing a boy, a cake, a train set, a balloon, and a toy car. While looking at this scene, they heard one of two sentences:

(1.1) The boy will eat the cake.

(1.2) The boy will move the cake.

In trials using (1.1), the subjects' eyes moved to the target object (the cake) sooner than in trials using (1.2) since the verb *eat* predicts that its object should be something edible, and the only edible thing in the scene is the cake. These experimental results "support a hypothesis in which sentence processing is driven by the predictive relationships between verbs, their syntactic arguments, and the real-world contexts in which they occur" (p. 247).

Experiments such as these—and many more along the same lines—provide human-oriented evidence in support of developing cognitive models of multisensory agent perception that centrally feature incremental analysis.[13] For more on the computational treatments of incrementality, see the deep dive in section 1.6.6.

1.4.3.3 Semantics Semantics—a word so big that it gives one pause. Most of this book can be viewed as a case study in defining what semantics is and how we can prepare agents to compute it. But for now, in this section on linguistic inspirations for agent development, let us focus on just two threads of scholarship in semantics: lexical semantics and formal semantics.[14]

Lexical semantics. Much of human knowledge about lexical semantics is reflected in human-oriented knowledge bases: lexicons, thesauri,[15] and wordnets (i.e., hierarchical

inventories of words that are organized conceptually rather than alphabetically). Although early practitioners of NLP held high hopes for the utility of machine-readable lexical knowledge bases, the disappointing reality is that human-oriented resources tend not to be well suited to computational aims. The main reason (for others, see the deep dive in section 1.6.7) is that, in order to effectively use such resources, people must bring to bear a lot of knowledge and reasoning about language and the world—all subconsciously, of course. To give just two examples: Have you ever attempted to use a thesaurus, or a large bilingual dictionary, for a language you are trying to learn? How do you choose a particular word or phrase among all those options? Similarly, have you ever tried to explain to a child why an unabridged dictionary needs a dozen senses to describe a seemingly simple word like *horse*? All this is so obvious to an adult native speaker—but not to a child, a nonnative speaker, or, even more so, a machine. So, for the enterprise of agent building, human-oriented scholarship in lexical semantics is most useful as a resource that computational linguists can consult when building knowledge bases specifically suited to machine processing. We will return to work of the latter sort in pillar 4.

Formal semantics. Formal semantics is a venerable area of study in linguistics and the philosophy of language that focuses primarily on three things: determining the truth conditions of declarative sentences; interpreting nondeclarative sentences on the basis of what would make the declarative variant true; and interpreting quantifiers. Of course, only a small part of language understanding actually involves truth conditions or quantification, which suggests that computational formal semantics cannot be considered an all-purpose approach to NLU. Moreover, truth judgments can only be made over unambiguous statements, which are rare in natural language. Intelligent agents certainly need to reason about truth, so formal semantics clearly has a role in agent functioning. But for that to happen, the NLU processes described in this book must first provide the prerequisite translation from natural language into an unambiguous metalanguage.

There does exist a branch of inquiry called computational formal semantics, which embraces the same topics as descriptive formal semantics and adds another: the use of theorem provers to determine the consistency of databases (Blackburn & Bos, 2005). We call it a *branch of inquiry* rather than (as yet) a field because (a) it assumes the abovementioned NLU-to-metalanguage translation prerequisite, and (b) some of the hottest issues turn out to be moot when subjected to the simple test of whether the problem actually occurs in natural language.

Regarding the latter, in his analysis of the place of formal semantics in NLP, Wilks (2011) reports a thought-provoking finding about a sentence type that has been discussed extensively in the theoretical literature, illustrated by the well-known example *John wants to marry a Norwegian*. Such sentences have been claimed to have two interpretations: John wants to marry a particular Norwegian (*de re*), and he wants to marry some Norwegian or other (*de dicto*). When Wilks carried out an informal web search for the corresponding "wants to marry a German" (since marrying a Norwegian was referenced in too many

linguistics papers), the first twenty hits all had the generic meaning, which suggests that if one wants to express the specific meaning, this turn of phrase is just not used. Wilks argues that computational semantics must involve both meaning representation and "concrete computational tasks on a large scale" (p. 7). He writes, "What is not real Compsem [computational semantics], even though it continues to masquerade under the name, is a formal semantics based on artificial examples and never, ever, on real computational and implemented processes" (p. 7).

This comment underscores two of the most important features that divide practitioners of NLP: judgments about the acceptable germination time between research results and practical utility, and the acceptable inventory of as-yet unfulfilled prerequisites. Formal semanticists who cast their work as computational assume a long germination time and require quite ambitious prerequisites to be fulfilled—most notably, a perfect language-to-metalanguage translation. However, they are attempting to treat difficult problems that will eventually need to be handled by human-level intelligent agents. The opposite point of view is that NLP is a practical pursuit that requires near-term results, within which long-term needs tend to be considered less central. The approach described in this book lies somewhere in between, pursuing a depth of analysis that has frequently been called ambitious but imposing firm requirements about computability.

Long germination time and outstanding prerequisites are not limited to formal semantics; they also apply to other research programs involving machine reasoning. Consider, for example, Winston's (2012) work on automating story understanding, which was further developed by Finlayson (2016). Winston's Genesis system carries out commonsense reasoning over stories, such as identifying that the concept of revenge plays a role in a story despite the absence of the word *revenge* or any of its synonyms. Finlayson's system, for its part, learns plot functions in folktales, such as villainy/lack, struggle and victory, and reward. A common thread of this reasoning-centric work is its reliance on inputs that are cleaner than everyday natural language. That is, like formal semanticists, these investigators press on in their study of reasoning, even though the prerequisite of automatic NLU remains outstanding.

Winston and Finlayson take different approaches to language simplification. Finlayson's learner requires semantically annotated texts, but the annotation process is only semiautomatic: it requires manual review and supplementation because the required features cannot be computed with high reliability given the current state of the art. These features include such things as the temporal ordering of events; mappings to WordNet senses; event valence—for example, the event's impact on the Hero; and the identification of dramatis personae, that is, character types.

Winston's system, for its part, takes as input plot summaries written in simple English. However, these are not typical plot summaries intended for people. Strictly speaking, these look more like logical forms with an English veneer. 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 excerpt includes both unexpected definitional components (essentially, elements of ontology) and a noncanonical use of the closed-class item *because* (in regular English, one would say *Estonia insulted Russia by relocating a war memorial*).

Our point is not that such inputs are inappropriate: they are very useful and entirely fitting in support of research whose focus lies outside the challenges of natural language as such. Our point is that these are excellent examples of the potential for dovetailing across research paradigms, with NLU of the type we describe here serving reasoning systems, and those reasoning systems, in turn, being incorporated into comprehensive agent systems.[16]

1.4.3.4 Pragmatics Pragmatic (also called *discourse* or *discourse-theoretic*) approaches attempt to explain language use holistically and, accordingly, can invoke all kinds of linguistic and nonlinguistic features. In this way, they are entirely in keeping with our methodology of agent development.

When pragmatics is approached from a descriptive, noncomputational perspective, it involves analyzing chunks of discourse using explanatory prose. The descriptions often invoke concepts—such as *topic*, *focus*, and *discourse theme*—that are understandable to people but have been difficult to concretize to the degree needed by computer systems. That is, when we read descriptive-pragmatic analyses of texts, our language-oriented intuitions fire and intuitively fill in the blanks of the associated pragmatic account.

Descriptive-pragmatic analyses tend to be cast as generalizations rather than rules that could be subjected to formal testing or hypotheses that could be overturned by counterevidence. So, the challenges in exploiting such analyses for computational ends are (a) identifying which generalizations can be made computer-tractable with what level of confidence and (b) providing agents with both the algorithms and the supporting knowledge to operationalize them.

Many of the microtheories we describe throughout the book involve pragmatics, as will become clear in our treatment of topics such as reference, ellipsis, nonliteral language, and indirect speech acts. In fact, it would not be an exaggeration to say that one of the core goals of Linguistics for the Age of AI is initiating a deep and comprehensive program of work on computational pragmatics.

One of the most widely studied aspects of pragmatics over the decades has been reference resolution. However, although individual insights can be quite useful for agent modeling, most approaches cannot yet be implemented in fully automatic systems because they require unobtainable prerequisites.[17] For example, prior knowledge of the discourse structure is required by the approaches put forth in Webber (1988, 1990) and Navarretta (2004). It is also required by Centering Theory (Grosz et al., 1995), which has been deemed computationally problematic and/or unnecessary by multiple investigators (e.g., Poesio, Stevenson, et al., 2004; Strube, 1998). Carbonell and Brown (1988), referring to Sidner (1981), say: "We … believe that dialog focus can yield a useful preference for anaphoric reference

selection, but lacking a computationally-adequate theory for dialog-level focus tracking (Sidner's is a partial theory), we could not yet implement such a strategy."

A new tradition of investigation into human cognition has been initiated by the field of computational psycholinguistics, whose practitioners are cognitive scientists looking toward statistical inference as a theoretically grounded explanation for some aspects of human cognition (e.g., Crocker, 1996; Dijkstra, 1996; Jurafsky, 2003; Griffiths, 2009). However, computational psycholinguistics relies on large corpora of manually annotated texts, whose scarcity limits progress, as it introduces a new aspect of the familiar knowledge bottleneck.

An obvious question is, Haven't aspects of pragmatics already been treated in computer systems? Yes, they have. (For deep dives into coreference, dialog act detection, and grounding, see sections 1.6.8–1.6.10.) However, these phenomena have been approached primarily using machine learning, which does not involve explanatory microtheories. Still, there is an associated knowledge angle that can, at least in part, be exploited in developing microtheories. Since most of the associated machine learning has been supervised, the methodology has required not only corpus annotation itself but the computational linguistic analysis needed to devise corpus annotation schemes. It cannot be overstated how much hard labor is required to organize a linguistic problem space into a manageable annotation task. This involves creating an inventory of all (or a reasonable approximation of all) eventualities; removing those that are too difficult to be handled by annotators consistently and/or are understood to be not treatable by the envisioned computer systems; and applying candidate schemes to actual texts to see how natural language can confound our expectations. Examples of impressive linguistic analyses of this genre include the MUC-7 coreference task description (Hirschman & Chinchor, 1997), the MUC-7 named-entity task description (Chinchor, 1997), the book-length manuscript on the identification and representation of ellipsis in the Prague Dependency Treebank (Mikulová, 2011), and the work on discourse-structure annotation described in Carlson et al. (2003).

Above we said that, within the realm of natural language processing, pragmatic phenomena have been addressed "primarily using machine learning." The word *primarily* is important, since there *are* some long-standing programs of research that address computational pragmatics from a knowledge-based perspective. Of particular note is the program of research led by Jerry Hobbs, which addresses many aspects of natural language understanding (e.g., lexical disambiguation; reference resolution; interpreting metaphors, metonymies, and compound nouns) using abductive reasoning with a reliance on world knowledge (e.g., Hobbs, 1992, 2004). An important strain of work in this area relates to studying the role of abductive inference in generating explanations of behavior, including learning (e.g., Lombrozo, 2006, 2012, 2016). Abduction-centered approaches to semantics, pragmatics, and agent reasoning overall are of considerable interest to cognitive systems developers (e.g., Langley et al., 2014). They are also compatible, both in spirit and in goals, with the program of NLU we present in this book. To make a sweeping (possibly, too

sweeping) generalization, the main difference between those programs of work and ours is one of emphasis: whereas Hobbs and Lombrozo focus on abduction as a logical method, we focus on treating the largest possible inventory of linguistic phenomena using hybrid analysis methods.

Continuing on the topic of language-related reasoning, one additional issue deserves mention: textual inference. Although at first blush it might seem straightforward to distinguish between what a text means and which inferences it supports, this can actually be quite difficult, as encapsulated by Manning's (2006) paper title, "Local Textual Inference: It's Hard to Circumscribe, But You Know It When You See It—and NLP Needs It." To take just one example from Manning, a person reading *The Mona Lisa, painted by Leonardo da Vinci from 1503–1506, hangs in Paris' Louvre Museum* would be able to infer that *The Mona Lisa is in France*. Accordingly, an NLP system with humanlike language processing capabilities should be able to make the same inference. However, as soon as textual inference was taken up by the NLP community as a "task," debate began about its nature, purview, and appropriate evaluation metrics. Should systems be provided with exactly the world knowledge they need to make the necessary inferences (e.g., Paris is a city in France), or should they be responsible for acquiring such knowledge themselves? Should language understanding be evaluated separately from reasoning about the world (if that is even possible), or should they be evaluated together, as necessarily interlinked capabilities? Should inferences orient around formal logic (*John has 20 dollars* implies *John has 10 dollars*) or naive reasoning (*John has 20 dollars* does not imply *John has 10 dollars*—because he has 20!)? Zaenen et al. (2005) and Manning (2006) present different points of view on all of these issues, motivated, as always, by differing beliefs about the proper scope of NLP, the time frame for development efforts, and all manner of practical and theoretical considerations.[18]

The final thing to say about pragmatics is that it is a very broad field that encompasses both topics that are urgently on agenda for intelligent agents and topics that are not. Good examples of the latter are three articles in a recent issue of *The Journal of Pragmatics* that discuss how/why doctors look at their computer screens (Nielsen, 2019); the use of underspecification in five languages, as revealed by transcripts of TED talks (Crible et al., 2019); and how eight lines of a playscript are developed over the course of rehearsals (Norrthon, 2019). Although all interesting in their own right, these topics are unlikely to make it to the agenda of AI in our lifetime. Our point in citing these examples is to illustrate, rather than merely state, the answer to a reasonable question: *With all the linguistics scholarship out there, why don't you import more?* Because (a) it is not all relevant (yet), and (b) little of it is importable without an awful lot of analysis, adjustment, and engineering.

1.4.3.5 Cognitive linguistics The recent growth of a paradigm called *cognitive linguistics* is curious with respect to its name because arguably all work on linguistics involves hypotheses about human cognition and therefore is, properly speaking, cognitive. However,

this is not the first time in the history of linguistics that a generic, compositional term has taken on a paradigm-specific meaning. After all, *theoretical linguistics* is commonly used as a shorthand for *generative grammar* in the Chomskian tradition, even though all schools of linguistics have theoretical underpinnings of one sort or another.

So, what *is* cognitive linguistics? If we follow the table of contents in Ungerer and Schmid's (2006) *An Introduction to Cognitive Linguistics*, then the major topics of interest for the field are prototypes and categories; levels of categorization; conceptual metaphors and metonymies; figure and ground (what used to be called topic/comment); frames and constructions; and blending and relevance. To generalize, what seems important to cognitive linguists is the world knowledge and reasoning we bring to bear for language processing, as well as the possibility of testing hypotheses on human subjects. From our perspective, all these topics are centrally relevant to agent modeling, but their grouping into a field called *cognitive linguistics* is arbitrary. To the extent that ongoing research on these topics produces descriptive content that can be made machine-tractable, this paradigm of work could be a contributor to agent systems.[19]

1.4.3.6 Language evolution A theoretical approach with noteworthy ripples of practical utility is the hierarchy of grammar complexity proposed by Jackendoff and Wittenberg (Jackendoff, 2002; Jackendoff & Wittenberg, 2014, 2017; hereafter referred to collectively as J&W). J&W emphasize that communication via natural language is, at base, a signal-to-meaning mapping. All the other levels of structure that have been so rigorously studied (phonology, morphology, syntax) represent intermediate layers that are not always needed to convey meaning.

J&W propose a hierarchy of grammatical complexity, motivating it both with hypotheses about the evolution of human language and with observations about current-day language use. They hypothesize that language evolved from a direct mapping between phonetic patterns and conceptual structures through stages that introduced various types of phonological, morphological, and syntactic structure—ending, finally, in the language faculty of modern humans. An early stage of language evolution—what they call *linear grammar*—had no morphological or syntactic structure, but the ordering of words could convey certain semantic roles following principles such as Agent First (i.e., refer to the Agent before the Patient). At this stage, pragmatics was largely responsible for utterance interpretation. As the modern human language faculty developed, it went through stages that introduced phrase structure, grammatical categories, symbols to encode abstract semantic relations (such as prepositions indicating spatial relations), inflectional morphology, and the rest. These enhanced capabilities significantly expanded the expressive power of the language system.

As mentioned earlier, the tiered-grammar hypothesis relates not only to the evolution of the human language faculty; it is also informed by phenomena attested in modern language use. Following Bickerton (1990), J&W believe that traces of the early stages of language

evolution survive in the human brain, manifesting when the system is either disrupted (e.g., by agrammatic aphasia) or not fully developed (e.g., in the speech of young children, and in pidgins). Expanding on this idea, J&W describe the human language faculty as "not a monolithic block of knowledge, but rather a palimpsest, consisting of layers of different degrees of complexity, in which various grammatical phenomena fall into different layers" (J&W, 2014, p. 67). Apart from fleshing out the details of these hypothesized layers of grammar, J&W offer additional modern-day evidence (beyond aphasia, the speech of young children, and pidgins) of the use of pre-final layers. For example:

1. Language emergence has been observed in two communities of sign language speakers (using Nicaraguan Sign Language and Al-Sayyid Bedouin Sign Language), in which the language of successive generations has shown increased linguistic complexity along the lines of J&W's layers.
2. The fully formed language called Riau Indonesian is structurally simpler than most modern languages. According to J&W (2014, p. 81), "the language is basically a simple phrase grammar whose constituency is determined by prosody, with a small amount of morphology."
3. The linguistic phenomenon of compounding in English can be analyzed as a trace of a pre-final stage of language development, since the elements of a compound are simply juxtaposed, with the ordering of elements suggesting the semantic head, and with pragmatics being responsible for reconstructing their semantic relationship.

What do language evolution and grammatical layers have to do with computational cognitive modeling? They provide theoretical support for independently motivated modeling strategies. In fact, one doesn't have to look to fringe phenomena like aphasia and pidgins to find evidence that complex and perfect structure is not always central to effective communication. We need only look at everyday dialogs, which are rife with fragmentary utterances and production errors—unfinished sentences, self-corrections, stacked tangents, repetitions, and the rest. All of this mess means that machines, like humans, must be prepared to apply far more pragmatic reasoning to language understanding than approaches that assume a strict syntax-to-semantics pipeline would expect.

Another practical motivation for preparing systems to function effectively without full and perfect structural analysis is that all that analysis is very difficult to perfect, and thus represents a long-term challenge for the AI community. As we work toward a solution, machines will have to get by using all the strategies they can bring to bear—not unlike a nonnative speaker, a person interpreting a fractured speech signal, or someone ramping up in a specialized domain. In short, whenever idealized language processing breaks down, we encounter a situation remarkably similar to the hypothesized early stages of language development: using word meaning to inform a largely pragmatic interpretation.

This concludes the necessarily lengthy explanation of the third part of our answer to the question, *What is Linguistics for the Age of AI?* It is the study of linguistics in service

of developing natural language understanding and generation capabilities (1) within an integrated, comprehensive agent architecture, (2) using human-inspired, explanatory modeling techniques, and (3) *leveraging insights from linguistic scholarship and, in turn, contributing to that scholarship.*

This whirlwind overview might give the impression that more of linguistic scholarship is *not* relevant than *is* relevant.[20] Perhaps. But that is not the main point. The main point is that a lot of it *is* relevant. Moreover, we are optimistic that practitioners in each individual field might be willing to think about how their results—even if not initially intended for AI—might be applied to AI, creating a cascade of effects throughout the scientific community. We find this a compelling vision for the future of AI and invite linguists to take up the challenge.

1.4.4 Pillar 4: All Available Heuristic Evidence Is Incorporated When Extracting and Representing the Meaning of Language Inputs

As we explained in pillar 2, agent modeling is most effective when (a) it is inspired by human functioning—to the extent that it can be modeled and is useful—and (b) it strongly emphasizes practicality. Since it is impossible to immediately achieve both depth and breadth of coverage of all phenomena using knowledge-based methods, it is, in principle, useful to import external sources of heuristic evidence—both knowledge bases and processors. However, as with exploiting linguistic scholarship, these importations come at a cost—often a high one that involves much more engineering than science. Both the decision-making about what to import and the associated work in the trenches are below the threshold of general interest and will not be discussed further in this book. Instead, we will simply describe some resources that have direct computational-linguistic relevance as examples of what's out there to serve agent systems as they progress toward human-level sophistication.

1.4.4.1 Handcrafted knowledge bases for NLP As discussed earlier, one of the main drawbacks of using human-oriented lexical resources for NLP is the machine's inability to contextually disambiguate the massively polysemous words of natural language. Accordingly, a core focus of attention in crafting resources expressly for NLP has been to provide the knowledge to support automatic disambiguation, which necessarily includes both syntactic and semantic expectations about heads and their dependents (most notably, verbs and the arguments they select). As George Miller rightly states, "Creating a handcrafted knowledge base is a labor-intensive enterprise that reasonable people undertake only if they feel strongly that it is necessary and cannot be achieved any other way" (Lenat et al., 1995). Quite a few reasonable people have seen this task as a necessity, taking different paths toward the same goal. By way of illustration, we briefly compare three handcrafted knowledge bases that were designed for use outside any particular language processing environment: the lexical databases called VerbNet and FrameNet and the ontology called Cyc.[21]

VerbNet (Kipper et al., 2006) is a hierarchical lexicon inspired by Levin's (1993) inventory of verb classes. The main theoretical hypothesis underlying Levin's work is that the similarity in syntactic behavior among the members of verb classes suggests a certain semantic affinity. Over the course of its development, VerbNet has expanded Levin's inventory to more than 200 verb classes and subclasses, increased the coverage to more than 4,000 verbs, and has described each class in terms of (a) argument structure, (b) legal syntactic realizations of the verb and its arguments, (c) a mapping of the verb to a WordNet synset (i.e., set of cognitive synonyms), and (d) an indication of coarse-grained semantic constraints on the arguments (e.g., human, organization).

FrameNet, for its part, was inspired by the theory of frame semantics (a version of construction grammar; Fillmore & Baker, 2009), which suggests that the meaning of most words is best described using language-independent semantic frames that indicate a type of event and the types of entities that participate in it. For example, an Apply_heat event involves a Cook, Food, and a Heating_instrument. A language-independent frame thus described can be evoked by given lexical items in a language (e.g., fry, bake). The FrameNet resource includes frame descriptions, words that evoke them, and annotated sentences that describe their use. Although FrameNet does include nouns as well as verbs, they are used mostly as dependents in verbal frames.

Apart from lexical knowledge bases, ontologies are also needed for knowledge-based AI, including but not limited to NLU (see, e.g., Guarino, 1998, for an overview). One of the largest and oldest ontology-building projects to date has been Cyc, whose goal is to encode sufficient commonsense knowledge to support any task requiring AI, including but not specifically oriented toward NLP. Doug Lenat, the project leader, described it as a "very long-term, high-risk gamble" (Lenat, 1995) that was intended to stand in contrast to what he called the "bump-on-a-log" projects occupying much of AI (see Stipp, 1995, for a nontechnical perspective). Although initially configured using the frame-like architecture typical of most ontologies—including all ontologies developed using Stanford's open-source Protégé environment (Noy et al., 2000)—the knowledge representation strategy quickly shifted to what developers call a "sea of assertions," such that each assertion is equally about each of the terms used in it. In a published debate with Lenat (Lenat et al., 1995), George Miller articulates some of the controversial assumptions of the Cyc approach: that commonsense knowledge is propositional; that a large but finite number of factual assertions (supplemented by machine learning of an as-yet undetermined type) can cover all necessary commonsense knowledge; that generative devices are unnecessary; and that a single inventory of commonsense knowledge can be compiled to suit any and all AI applications.[22] Additional points of concern include how people can be expected to manipulate (find, keep track of, detect lacunae in) a knowledge base containing millions of assertions, and the ever present problem of lexical ambiguity. Yuret (1996) offers a fair-minded explanatory review of Cyc in the context of AI.

Before closing this section, we must mention the Semantic Web, which is another source of manually encoded data intended to support the machine processing of text. This time, however, the data is in the form of tags that serve as metadata on internet pages. The Semantic Web vision arose from the desire to make the content of the World Wide Web more easily processed by machines. Berners-Lee et al. (2001) write: "The Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users." In effect, the goal was to transform the World Wide Web into a richly annotated corpus, but in ways that remain largely unspecified (see Sparck Jones, 2004, for an insightful critique).

We refer to the Semantic Web as a vision rather than a reality because work toward automatically annotating web pages, rather than manually providing the annotations, has been largely sidelined by the Semantic Web community in favor of creating formalisms and standards for encoding such meaning, should it ever become available. Moreover, even the simpler desiderata of the Semantic Web community—such as the use of consistent metadata tags—are subject to heavy real-world confounders. Indeed, metadata, which is typically assumed to mean manually provided annotations realized by hypertext tags, is vulnerable to inconsistency, errors, laziness, intentional (e.g., competition-driven) falsification, subconscious biases, and bona fide alternative analyses. Standardization of tags has been a topic of intense discussion among the developers, but it is not clear that any practical solution to this problem is imminent. As a result, especially in critical applications, the metadata cannot currently be trusted.

While the current R&D paradigm of the Semantic Web community might ultimately serve some intelligent agents—particularly in applications like e-commerce, in which true language understanding is not actually needed (cf. Uschold, 2003)—use of the term *Semantic Web* to describe the work is unfortunate since automatically extracting meaning is centrally absent. As Shirky (2003) writes in his entertaining albeit rather biting analysis, "The Semantic Web takes for granted that many important aspects of the world can be specified in an unambiguous and universally agreed-on fashion, then spends a great deal of time talking about the ideal XML formats for those descriptions. This puts the stress on the wrong part of the problem." In sum, when viewed from the perspective of developing deep NLU capabilities, the web—with or without metadata tags—is simply another corpus whose most challenging semantic issues are the same as for any corpus: lexical disambiguation, ellipsis, nonliteral language, implicature, and the rest.

1.4.4.2 Using results from empirical NLP Empirical NLP has had many successes, demonstrating that certain types of language-related tasks are amenable to statistical methods. (For an overview of empirical NLP, see the deep dive in section 1.6.11.) For example, machine translation has made impressive strides for language pairs for which sufficiently large parallel corpora are available; syntactic parsers for many languages do a pretty good

job on the more canonical text genres; and we all happily use search engines to find what we need on the internet. The task for developers of agent systems, then, is to identify engines that can provide useful heuristic evidence for NLU, no matter how this evidence is obtained.[23]

The most obvious source of useful heuristics are preprocessors and syntactic parsers, which have historically been among the most studied topics of NLP. Syntax being as tricky as it is—particularly in less formal genres—parsing results remain less than perfect.[24] However, when such results are treated as overridable heuristic evidence within a semantically oriented language understanding system, they can still be quite useful.

Another success from the statistical paradigm that can be broadly applied to agent systems is case role labeling. Case roles—otherwise known as semantic roles—indicate the main participants in an event, such as the agent, theme, instrument, and beneficiary. In a knowledge-lean environment, these roles are used to link uninterpreted text strings; so the semantics in this approach is in the role label itself. If a semantic role–labeling system is provided with a set of paraphrases, it should be able to establish the same inventory of semantic role assignments for each.[25] For example, given the sentence set

- Marcy forced Andrew to lend her his BMW.
- Andrew was forced by Marcy into lending her his BMW.
- Andrew lent his BMW to Marcy because she made him.

a semantic role labeler should recognize that there is a *lend* event in which Andrew is the agent, his BMW is the theme, Marcy is the beneficiary, and Marcy caused the event to begin with.

Semantic role–labeling systems (e.g., Gildea & Jurafsky, 2002) are typically trained using supervised machine learning, relying on the corpus annotations provided in such resources as PropBank (Palmer et al., 2005) and FrameNet (Fillmore & Baker, 2009). Among the linguistic features that inform semantic role labelers are the verb itself, including its subcategorization frame and selectional constraints; aspects of the syntactic parse tree; the voice (active vs. passive) of the clause; and the linear position of elements. As Jurafsky and Martin (2009, pp. 670–671) report, semantic role–labeling capabilities have improved system performance in tasks such as question answering and information extraction.

Coreference resolution within statistical NLP has also produced useful results, though with respect to a rather tightly constrained scope of phenomena and with variable confidence across different referring expressions, as we detail in chapter 5.

Distributional semantics is a popular statistical approach that operationalizes the intuitions that "a word is characterized by the company it keeps" (Firth, 1957) and "words that occur in similar contexts tend to have similar meanings" (Turney & Pantel, 2010).[26] Distributional models are good at computing similarities between words. For example, they can establish that *cat* and *dog* are more similar to each other than either of these is to *airplane*, since *cat* and *dog* frequently co-occur with many of the same words: *fur*, *run*,

owner, *play*. Moreover, statistical techniques, such as Pointwise Mutual Information, can be used to detect that some words are more indicative of a word's meaning than others. For example, whereas *fur* is characteristic of dogs, very frequent words like *the* or *has*, which often appear in texts with the word *dog*, are not.

Although distributional semantics has proven useful for such applications as document retrieval, it is not a comprehensive approach to computing meaning since it only considers the co-occurrence of words. Among the things it does not consider are

- the ordering of the words, which can have profound semantic implications: *X attacked Y* versus *Y attacked X*;
- their compositionality, which is the extent to which the meaning of a group of words can be predicted by the meanings of each of the component words; for example, in most contexts, *The old man kicked the bucket* has nothing to do with the physical act of kicking a cylindrical open container;[27] and
- any of the hidden sources of meaning in language, such as ellipsis and implicature.

To sum up, syntactic parsing, semantic role labeling, coreference resolution, and distributional semantics exemplify ways in which empirical NLP can serve NLU. We do not, however, expect empirical methods to have similar successes in more fundamental aspects of semantics or pragmatics. As Zaenen (2006) explains, annotating semantic features is significantly more difficult than annotating syntactic features; accordingly, related annotation efforts to date have reflected substantial simplifications of the real problem space. Moreover, even if semantic annotation were possible, it is far from clear that the learning methods themselves would work very well over a corpus thus annotated since the annotations will necessarily include meanings not overtly represented by text strings. (For more on corpus annotation, see the deep dive in section 1.6.12.)

This concludes the fourth part of our answer to the question, *What is Linguistics for the Age of AI?* It is the study of linguistics in service of developing natural language understanding and generation capabilities (1) within an integrated, comprehensive agent architecture, (2) using human-inspired, explanatory modeling techniques, (3) leveraging insights from linguistic scholarship and, in turn, contributing to that scholarship, and (4) *incorporating all available heuristic evidence when extracting and representing the meaning of language inputs*.

1.5 The Goals of This Book

The cognitive systems–inspired, computer-tractable approach to NLU described here has been under continuous development, with various emphases, for over thirty-five years. This time frame is noteworthy because the program of work began when computational linguistics and knowledge-based approaches were still considered a proper part of NLP, when AI

was not largely synonymous with machine learning, and when words like *cognitive*, *agents*, and *ontology* were not yet commonplace in the popular press.

A good question is why this program of work has survived despite finding itself outside the center of attention of both mainstream practitioners and the general public. The reason, we believe, is that the vision of human-level AI remains as tantalizing now as when first formulated by the founders of AI over a half century ago. We agree with Marvin Minsky that "We have got to get back to the deepest questions of AI and *general* intelligence and quit wasting time on little projects that don't contribute to the main goal. We can get back to them later" (quoted in Stork, 1997, p. 30). It is impossible to predict how long it will take to attain high-quality NLU, but John McCarthy's estimate about AI overall, as reported by Minsky, seems appropriate: "If we worked really hard we'd have an intelligent system in from four to four hundred years" (Stork, p. 19).

Witticisms aside, endowing LEIAs with the ability to extract an iceberg of meaning from the visible tip reflected by the words in a sentence is not a short-term endeavor. At this point in history, it more properly belongs to the realm of science than technology, although we can and have packaged useful results for particular tasks in specific domains. Accordingly, the main contribution of the book is scientific. We present a theory of NLU for LEIAs that includes its component algorithms and knowledge resources, approaches for extending the latter, and a methodology of its integration with the extralinguistic functionalities of LEIAs. The theory can be applied, in full or in part, to any agent-based system. Viewed this way, our contribution must be judged on how well it stands the test of time, how effectively it serves as a scaffolding for deeper exploration of the component phenomena and models, and how usefully it can be applied to any of the world's languages.

While our main emphasis is on science, engineering plays an important role, too. Much of what we describe has already been implemented in systems. We believe that implementation is essential in cognitively inspired AI to ensure that the theories can, in fact, serve as the basis for the development of applications. When we say that a LEIA *does X*, it means that algorithms have been developed to support the behavior. Many of these algorithms have already been included in prototype application systems. Others are scheduled for inclusion, as our team continues system-development work.

The language descriptions and algorithms presented here cover both generic theoretical and specific system-building aspects. They are specific in that they have been developed within a particular theoretical framework (Ontological Semantics), which has been implemented in a particular type of intelligent agents (LEIAs) in a particular cognitive architecture (OntoAgent). In this sense, the work is real in the way that system developers understand. On the other hand, the descriptions and algorithms reflect a rigorous analysis of language phenomena that is valid outside its association with this, or any other, formalism or application environment.

Descriptions of complex phenomena in any scientific realm have a curious property: the better they are, the more self-evident they seem. Linguistic descriptions are particularly subject to such judgments because every person capable of reading them has functional

expertise in language—something that cannot be said of mathematics or biology. Even within the field of linguistics, rigorous descriptions of how things work—the kind you need, for example, if you have ever tried to fully master a foreign language—are traditionally unpublishable unless they are subsumed under some theoretical umbrella. This is unfortunate as it leaves an awful lot of work for computational linguists to do.

As we have explained, published linguistic scholarship is suitable only as a starting point for the knowledge engineering required to support language processing in LEIAs. Grammar books leave too much hidden behind lists flanked by *e.g.* and *etc.*; discourse-theoretic accounts regularly rely on computationally intractable concepts such as *topic* and *focus*; and lexical resources intended for people rely on people's ability to, for example, disambiguate the words used in definitions and recognize the nuances distinguishing near-synonyms. Artificial intelligent agents do not possess these language processing and reasoning abilities, so linguistic resources aimed at them must make all of this implicit information explicit. It would be a boon to linguistics overall if the needs of intelligent agents spurred a proliferation of precise, comprehensive, and computer-tractable linguistic descriptions. As this has not been happening, our group is taking on this work, albeit at a scale that cannot rival the output potential of an entire field.

What we hope to convey in the book is how a knowledge-based, deep-semantic approach to NLU works, what it can offer, and why building associated systems is not only feasible but necessary. Naturally, the composition of actual agent system prototypes will vary, as it will reflect different theoretical, methodological, and tactical decisions. However, all such systems will need to account for the same extensive inventory of natural language phenomena and processes that we address in this book.

A note on how to read this book. There is no single best, straight path through describing a large program of work, including its theoretical and methodological substrates, its place in the history of the field, and its plethora of technical details. Readers will inevitably have different most-pressing questions arising at different points in the narrative. We, therefore, make three tactical suggestions:

- If something is not immediately clear, read on; a clarification might be just around the corner.
- Skip around liberally, using the table of contents as your guide.
- Understand that some repetition in the narrative is a feature, not a bug, to help manage the reader's cognitive load.

1.6 Deep Dives

1.6.1 The Phenomenological Stance

We are interested in modeling the agents from the first-person, phenomenological perspective.[28] This means that each agent's knowledge, like each person's knowledge, is assumed

at all times to be incomplete and potentially at odds with how the world really is (i.e., it can contrast with the knowledge of a putative omniscient agent, which would embody what's known as the third-person perspective). To borrow a term from ethology, we model each LEIA's *umwelt*.

We have demonstrated the utility of modeling agents from multiple perspectives by implementing and testing non-toy computational models of both first-person and third-person (omniscient) agents in application systems. For example, the Maryland Virtual Patient (MVP) system (see chapter 8) featured an omniscient agent endowed with an expert-derived, state-of-the-art explanatory model of the physiology and pathology of the human esophagus, as well as clinical knowledge about the experiences of humans affected by esophageal diseases. This omniscient agent (a) ensured the realistic progression of a virtual patient's disease and healing processes, in response to whatever interventions were selected by system users, and (b) provided ground-truth knowledge to the tutoring agent who was not, however, omniscient: like any physician, it had access only to that subset of patient features that had either been reported by the patient or were returned as test results. The virtual patients in the system were, likewise, modeled from the first-person perspective: they were endowed with different partial, and sometimes objectively incorrect, knowledge. Importantly for this book, the virtual patients could expand and correct their knowledge through experiences and interactions with the human trainees, who played the role of attending physicians. For example, virtual patients were shown to be able to learn both ontological concepts and lexical items through conversation with the human trainees.[29]

Another human-inspired aspect of our modeling strategy is the recognition that the agents' knowledge can be internally contradictory and/or vague. For example, in a recent robotic application the agent was taught more than one way to perform a complex task through dialogs with different human team members (see section 8.4). In any given system run, the agent carried out the task according to the instructions that it had learned from the team member participating in that run. When asked to describe the task structure, the agent offered all known options: "According to A, the complex task is T_1; while according to B, it is T_2."

To sum up our phenomenological stance, we model intelligent agents to operate on the basis of folk psychology—that is, their view of the world (like a human's) is less than scientific. Each human and artificial member of the society is expected to have different first-person perspectives, but they have sufficient overlap to support successful communication and joint operation. Incompatibilities and lacunae in each agent's knowledge are expected to occur. One of the core methods of eliminating incompatibilities and filling lacunae is through natural language communication.

1.6.2 Learning

The ability to understand language is tightly coupled with the ability to learn. As emphasized earlier, in order to understand language, people must possess a lot of knowledge, and that knowledge must be learned. In developing artificial intelligent agents, learning can be either delegated to human knowledge acquirers (whose job description has been more or less the same since the 1970s) or modeled as an automatic capability of agents. Since modeling humanlike behavior is a core requirement for LEIAs, they, like people, must be able to learn using natural language.

The core prerequisite for language-based learning—be it through reading, being taught, or participating in nonpedagogically oriented dialogs—is the ability to understand natural language. But, as we just pointed out, that process itself requires knowledge! Although this might appear to be a vicious circle, it is actually not, as long as the agent starts out with a critical mass of ontological and lexical knowledge, as well as the ability to bootstrap the learning process—by generating meaning representations, using reasoning engines to make inferences, managing memory, and so on. Focusing on bootstrapping means that we are not modeling human learning as if it were from scratch—particularly since, for human brains, there arguably is no scratch. Of all the types of learning that LEIAs must, and have in the past, undertaken, we will focus here on the learning of new words and new facts—that is, new propositional content recorded as ontologically grounded meaning representations.[30]

1.6.3 NLP and NLU: It's Not Either-Or

Over the past three decades, the ascendance of the statistical paradigm in NLP and AI in general has seen knowledge-based methods being variously cast as outdated, unnecessary, lacking promise, or unattainable. However, the view that a competition exists between the approaches is misplaced and, upon closer inspection, actually rather baffling. This should become clear as we walk through some unmotivated beliefs that, by all indications, are widely held in the field today.[31]

Unmotivated belief 1. *There is a knowledge bottleneck and it affects only knowledge-based approaches*. Although knowledge-lean approaches purport to circumvent the need for manually acquired knowledge, 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 semantics and pragmatics (see section 1.6.12). 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? Issues that must be solved to attain the next level of quality in automatic language processing.

Unmotivated belief 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 twenty-five years had remained with the goal of automating language understanding.

Unmotivated belief 3. *NLU is an extension of NLP.* Fundamental NLU has little to nothing in common with current mainstream NLP; in fact, it has much more in common with robotics. Like robotics, NLU is currently most fruitfully 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 often criticized for not being immediately applicable to all domains (under the pressure of evaluation frameworks entrenched in statistical NLP). 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 shallow semantic or pragmatic features. Of course, marrying robotics with NLU is a natural fit.

Unmotivated belief 4. *It's either NLP or NLU.* One key to the success of NLP has been finding applications and system configurations 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 two 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."[32] However, not all language-oriented applications offer such remarkable simplifications. For example, agents in dialog systems receive one and only one formulation of each utterance. Moreover, they must 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. (DuBois et al., 2000–2005)[33]

Unsupported by the visual context or the intonation 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 (we 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 belief 5. *Whereas mainstream NLP is realistic, deep NLU is unrealistic*. This faulty assessment seems to derive from an undue emphasis on compartmentalization. If one plucks NLU out of overall agent cognition and expects meaning analysis to be carried out to perfection in isolation from world and situational knowledge, then, indeed, the task is unrealistic. However, this framing of the problem is misleading. To understand language inputs, a cognitive agent must know what kinds of information to rely on during language analysis and why. It must also use a variety of kinds of 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 worth 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 implicatures 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."*

One additional aspect of the realistic/unrealistic assessment must be mentioned. A large portion of work on supervised learning in support of NLP has been carried out under less than realistic conditions. Task specifications normally include in their purview only the simpler instances of the given phenomenon, and manually annotated corpora are often provided to developers for both the training and the evaluation stages of system development. This means that the systems configured according to such specifications cannot perform at their evaluated levels on raw texts (for discussion, see Mitkov, 2001, and chapter 9). To generalize, judgments about feasibility cannot be made in broad strokes at the level of statistical versus knowledge-based systems.

To recap, we 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.

1.6.4 Cognitive Systems: A Bird's-Eye View

To assess the current views on the role of NLP in computational cognitive science, we turn to an authoritative survey of research in cognitive architectures and their associated cognitive systems (Langley et al., 2009). The survey analyzes nine capabilities that any good cognitive architecture must have: (1) recognition and categorization, (2) decision-making and choice, (3) perception and situation assessment, (4) prediction and monitoring, (5) problem solving and planning, (6) reasoning and belief maintenance, (7) execution and action, (8) interaction and communication, and (9) remembering, reflection, and learning. Langley et al. primarily subsume NLP under *interaction and communication* but acknowledge that it involves other aspects of cognition as well. The following excerpt summarizes their view. We have added indices in square brackets to link mentioned phenomena with the aspects of cognition just listed:

> A cognitive architecture should … support mechanisms for transforming knowledge into the form and medium through which it will be communicated [8]. The most common form is … language, which follows established conventions for semantics, syntax and pragmatics onto which an agent must map the content it wants to convey. … One can view language generation as a form of planning [5] and execution [7], whereas language understanding involves inference and reasoning [6]. However, the specialized nature of language processing makes these views misleading, since the task raises many additional issues. (Langley et al., 2009)

Langley et al.'s (2009) analysis underscores a noteworthy aspect of most cognitive architectures: even if reasoning is acknowledged as participating in NLP, the architectures are modularized such that core agent reasoning is separate from NLP-oriented reasoning. This perceived dichotomy between general reasoning and reasoning for NLP has been influenced by the knowledge-lean NLP paradigm, which both downplays reasoning as a tool for NLP and uses algorithms that do not mesh well with the kind of reasoning carried out in most cognitive architectures. However, if NLP is pursued within a knowledge-based paradigm, then there is great overlap between the methods and knowledge bases used for all kinds of agent reasoning, as well as the potential for much tighter system integration. Even more importantly, language processing is then, appropriately, not relegated to the input-output periphery of cognitive modeling because reasoning about language is a core task of a comprehensive cognitive model.

Consider, for example, an architecture in which verbal action is considered not separate from other actions (as in Langley et al.'s [2009] point [7] vs. point [8]) but simply another class of action. Such an organization would capture the fact that, in many cases, the set of plans for attaining an agent's goal may include a mixture of physical, mental, and verbal

actions. For example, if an embodied agent is cold, it can ask someone else to close the window (a verbal action), it can close the window itself (a physical action), or it can focus on something else so as not to notice its coldness (a mental action). Conversely, one and the same element of input to reasoning can be generated from sensory, language, or interoceptory (i.e., resulting from the body's signals, e.g., pain) input or as a result of prior reasoning. For example, a simulated embodied agent can choose to put the goal "have cut not bleed anymore" on its agenda—with an associated plan like "affix a bandage"—because it independently noticed that its finger was bleeding; because someone pointed to its finger and then it noticed it was bleeding (previously, its attention was elsewhere); because someone said, "Your finger is bleeding"; or because it felt pain in its finger and then looked and saw that it was bleeding.

The conceptual and algorithmic frameworks developed in the fields of agent planning, inference, and reasoning can all be usefully incorporated into the analysis of the semantics and pragmatics of discourse. For example, the pioneering work of Cohen, Levesque, and Perrault (e.g., Cohen & Levesque, 1990; Perrault, 1990) demonstrated the utility of approaching NLP tasks in terms of AI-style planning; planning is a first-order concern in the field of natural language generation (e.g., Reiter, 2010); and inference and reasoning have been at the center of attention of AI-style NLP for many years.

Returning to Langley et al.'s (2009) survey, their section on open issues in cognitive architectures states: "Although natural language processing has been demonstrated within some architectures, few intelligent systems have combined this with the ability to communicate about their own decisions, plans, and other cognitive activities in a general manner." Indeed, of the eighteen representative architectures briefly described in the appendix, only two—SOAR (Lewis, 1993) and GLAIR (Shapiro & Ismail, 2003)—are overtly credited with involving NLP, and one, ACT-R, is credited indirectly by reference to applied work on tutoring (Koedinger et al., 1997) within its framework. Although many cognitive architectures claim to have implemented language processing (thirteen of the twenty-six included in a survey by Samsonovich, http://bicasociety.org/cogarch/architectures.pdf), most of these implementations are limited in scope and depth, and none of them truly has language at the center of its scientific interests.

The LEIAs we describe throughout the book pursue deep NLU within the cognitive systems paradigm. Of the few research programs worldwide that currently pursue similar aims, perhaps the closest in spirit are those of Cycorp and the University of Rochester's TRAINS/TRIPS group (Allen et al., 2005). We will not attempt point-by-point comparisons with these because in order for such comparisons to be useful—rather than nominal, box-checking exercises—heavy preconditions must be met, both in the preparation and in the presentation.[34] In addition, the differences between research programs are certainly largely influenced by nonscientific considerations that live as explanatory folklore in actual research operations: which research projects were funded, which dissertations were written, which goals were prioritized for which reasons, and so on. In short, any investigator who is interested in a head-to-head comparison will have a particular goal in mind, and it is that goal that will delimit and make useful the process of drawing comparisons.

As concerns cognitive systems that include deep natural language processing but without an emphasis on fundamentally advancing our understanding of language processing, two noteworthy examples are the robotic systems reported by Lindes and Laird (2016) and Scheutz et al. (2017). The former system implements a parser based on embodied construction grammar (Feldman et al., 2009). The latter system uses an algorithm by which a "Lambda calculus representation of words could be inferred in an inverse manner from examples of sentences and their formal representation" (Baral et al., 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 of grounding linguistic expressions in the robot's world model. As a result of the above choice, both the robot's language processing capabilities and its conceptual knowledge cover the minimum necessary for immediate system needs. However, if the ultimate goal is to develop robotic language understanding that approaches human-level sophistication, then the large number of linguistic issues addressed in this book cannot be indefinitely postponed.

1.6.5 Explanation in AI

The ability to explain behavior in human terms is not a forte of the current generation of AI systems. The following statement by Rodney Brooks (2015) provides a good illustration of the current state of the art in a representative AI application:

> Today's chess programs have no way of saying why a particular move is "better" than another move, save that it moves the game to a part of a tree where the opponent has less good options. A human player can make generalizations and describe why certain types of moves are good, and use that to teach a human player. Brute force programs cannot teach a human player, except by being a sparring partner. It is up to the human to make the inferences, the analogies, and to do any learning on their own. The chess program doesn't know that it is outsmarting the person, doesn't know that it is a teaching aid, doesn't know that it is playing something called chess nor even what "playing" is. Making brute force chess playing perform better than any human gets us no closer to competence in chess. (p. 109).

For an agent to serve as a true AI—meaning an equal member of a human-agent team—it must be able to generate explanations of its behavior that are elucidating and satisfying to people.

The need for explanation in AI has certainly been recognized, as evidenced, for example, by the existence of DARPA's Explainable AI program. A workshop on the topic was held at IJCAI-2017. This is a positive development. Constructing explanations is not an easy task. Constructing relevant explanations is an even more difficult one. It seems that very few things can demonstrate that an artificial intelligent agent possesses at least a vestige of human-level intelligence as well as its ability to generate explanations specifically

for a particular audience and state of affairs in the world. Without these constraints, many explanations, while being technically accurate, might prove unedifying or inappropriate. Plato's reported definition of humans as "featherless bipeds" may have engendered Diogenes's witty and cynical response (according to Diogenes Laertius, Diogenes the Cynic plucked feathers off a chicken and presented it to Plato as a counterexample) but will not be treated by most people in most situations as an enlightening characterization.

Explanations differ along multiple parameters. For example, the basis of an explanation can be empirical or causal. Empirical explanations can range from "have always done it this way and succeeded" to appeals to authority ("this is what my teammate told me to do"). Causal explanations can appeal to laws of physics/biology or to folk psychology ("because people tend to like people they have helped"). And causes themselves may be observable ("the table is set for dinner because I just saw Zach setting it") or unobservable ("Bill is silent because he does not know the answer to the question I asked").

To provide explanations for unobservables, intelligent agents must be equipped with a theory of mind, which is the ability to attribute mental states (beliefs, desires, emotions, attitudes) to oneself and others. Operationalizing the twin capabilities of metacognition (the analysis of self) and mindreading (the analysis of others) is facilitated by organizing the agent's models of self and others in folk-psychological terms (see Carruthers, 2009, for a discussion of the interaction between mindreading and metacognition). Agents able to understand their own and others' behavior in folk-psychological terms will be able to generate humanlike explanations and, as a result, be better, more trusted, collaborators.

The ability to explain past behavior in terms of causes, and future behavior in terms of expected effects, is needed not only to support interpersonal interactions but also for language understanding itself. For example, indirect speech acts ("I'd be much happier if I didn't have to cook tonight") require the listener to figure out why the speaker said what he or she said, which is a prerequisite for selecting an appropriate response. This means that, although explanation has traditionally been treated separately from NLU, this separation cannot be maintained: a model of explanation must be a central part of the NLU module itself. And, since there do not exist any behavior-explanation reasoners that we can import—and since we do not rely on unavailable prerequisites—developing associated reasoning capabilities is necessarily within our purview.

On the practical level, the agent models we build are explanatory not only because their operation is interpretable in human folk-psychological terms but also because our systems' internal workings—static knowledge, situational knowledge, and all algorithms—are inspectable by people (though familiarity with the formalism is, of course, required).

Philosophers and psychologists (Woodward, 2019; Lombrozo, 2006, 2016) have devoted significant attention to the varieties and theories of explanation, often coming to unexpected conclusions, as when Nancy Cartwright (1983) persuasively argues that, despite their great explanatory power, fundamental scientific laws are not descriptively adequate—that is, they

do not describe reality. The corollary for us is that the scientific view of the world is different from the view of the world reflecting everyday human functioning. We believe that our task is to develop LEIAs that are primarily intended to model and interact with these everyday human agents. Such agents have much broader applicability in all kinds of practical applications than agents that are omniscient, whether in a given field or across fields.

A related issue is whether to endow LEIAs with normative or descriptive rationality. Normative rationality describes how people *should* make decisions, whereas descriptive rationality describes how they actually *do* make decisions. In their discussion of human and artificial rationality, Besold and Uckelman (2018) persuasively argue that "humans do not, generally, attain the normative standard of rationality" proposed in philosophy and cognitive science. As a corollary, a LEIA endowed with normative rationality will behave in ways that people will not interpret as sufficiently humanlike. This state of affairs evokes the concept of "the uncanny valley" (Mori, 2012). Indeed, Besold and Uckelman continue: "Because humans fall short of perfect rationality, a perfectly rational machine would almost immediately fall victim to the uncanny valley." Their solution is to base agents' theory of mind and mind-reading capabilities not on normative rationality but on descriptive rationality—that is, on how people actually act rather than how they say they are supposed to act. We choose to model descriptive rationality and ground explanations in folk psychology. Such explanations are not necessarily scientific, nor necessarily (always) true, but we see to it that they are always contextually appropriate and that they take into account the goals, plans, biases, and beliefs of both the producer and the consumer of the explanation.

To summarize, models of explanation in Linguistics for the Age of AI rely on the folk-psychological capabilities of mindreading and metacognition because the people who will interact with—and, with any luck, ultimately trust—AI systems need explanations in terms that they understand and find familiar.[35]

1.6.6 Incrementality in the History of NLP

For any task—from speech recognition to syntactic parsing to full natural language understanding—one can implement any or all component processors using any degree of incrementality. Ideally, the incremental (sub)systems would correctly process every incoming chunk of input and seamlessly add to the overall analysis, as fragments turned into sentences and sentences into discourses. However, defining *chunk* is anything but obvious: Is a chunk a word? A phrase? A clause? Must the optimal chunk size be dynamically calculated depending on the input? Can the system backtrack and change its analysis (i.e., be non-monotonic) or is it permitted only to add to previously computed analyses (i.e., be monotonic)? Is it better to wait for larger chunks in order to achieve higher initial accuracy or, as in automatic speech recognition systems, must the system decide fast and finally?

Köhn (2018) illustrates the challenges of incrementality in his analysis of the Verbmobil project (e.g., Wahlster, 2000), which aimed at developing a portable, simultaneous speech-to-speech translation system. He writes:

> The project developed speech recognition and synthesis components, syntactic and semantic parsers, self-correction detection, dialogue modeling and of course machine translation, showing that incrementality is an aspect that touches nearly all topics of NLP. This project also exemplifies that building incremental systems is not easy, even with massive funding [equivalent to approximately 78 million Euros when adjusted for inflation]: Only one of the many components ended up being incremental and the final report makes no mention of *simultaneous* interpretation. (p. 2991)

One way to incorporate incrementality into NLP systems is to focus on a narrow domain in which the focus is not on the coverage of linguistic phenomena but on the holistic nature of the application. For example, Kruijff et al. (2007) and Brick and Scheutz (2007) report robotic systems with broadly comparable cognitive architectures and capabilities. For the purposes of our language-centric overview, these programs of work are similar in that they acknowledge the necessity of language understanding and integrate related capabilities into the overall robotic architecture, but without taking on all of the challenges of unconstrained language use. For example, Kruijff et al. have a dialog model, they ground the incremental interpretation in the overall understanding of the scene, and they bunch as-yet ambiguous interpretations into what they call a *packed* representation, which represents all information shared by alternative analyses just once. However, their robot's world contains only three mugs and a ball, and utterances are limited to basic assertions and commands related to those entities, such as "the mug is red" and "put the mug to the left of the ball." So, whereas some necessary components of a more sophisticated language processing system are in place, the details of realistic natural language have not yet been addressed.

Another system that belongs to this narrow-domain category is the one described in DeVault et al. (2009).[36] It can predict at which point in a language stream it has achieved the maximum understanding of the input and then complete the utterance. For example, given the utterance "We need to," the system offers the completion "move your clinic"; given the utterance "I have orders," the system offers the completion "to move you and this clinic." Presumably, these continuations can be made confidently because the domain-specific ontology and task model offer only one option for each utterance continuation. The method employed involved machine learning using 3,500 training examples that were mapped into one of 136 attribute-value matrix frames representing semantic information in the ontology and task model.

A computational model of pragmatic incrementality is presented in Cohn-Gordon et al. (2019). Among the goals of their model is to account for the fact that people make anticipatory implicatures partway through utterances (cf. Sedivy, 2007). For example, if shown a scene with a tall cup, a short cup, a tall pitcher, and a key, a listener who hears "Give me the tall __" will fixate on the tall cup before the utterance is complete, since the only reason to use *tall* would be to distinguish between cups; since there is only one pitcher, there is no need to refer to its height. This model assigns a probability preference to *cup* (over *pitcher*) when the word *tall* is consumed, which formally accounts for the implicature.

However, this implicature is cancelable: if the utterance actually ends with *pitcher*, all referents apart from the pitcher are excluded.

1.6.7 Why Machine-Readable, Human-Oriented Resources Are Not Enough

The 1980s and early 1990s showed a surge of interest in automatically extracting NLP-oriented knowledge bases from the newly available machine-readable dictionaries as a means of overcoming the knowledge bottleneck. This research was based on two assumptions: (a) that machine-readable dictionaries contain information that is useful for NLP and (b) that this information would be relatively easy to extract into a machine-oriented knowledge base (Ide & Véronis, 1993). For example, it was expected that an ontological subsumption hierarchy could be extracted using the hypernyms that introduce most dictionary definitions (*a dog is a domesticated carnivorous mammal*) and that other salient properties could be extracted as well (*a dog ... typically has a long snout*). Although information in an idealized lexicon might be both useful and easy to extract, actual dictionaries built *by* people *for* people require human levels of language understanding and reasoning to be adequately interpreted. For example:

1. Senses are often split too finely for even a person to understand why.
2. Definitions regularly contain highly ambiguous descriptors.
3. Sense discrimination is often left to examples, meaning that the user must infer the generalization illustrated by the example.
4. The hypernym that typically begins a definition can be of any level of specificity (a dog is an animal/mammal/carnivore/domesticated carnivore), which confounds the automatic learning of a semantic hierarchy.
5. The choice of what counts as a salient descriptor is variable across entries (dog: a domesticated carnivorous mammal; turtle: a slow-moving reptile).
6. Circular definitions are common (a tool is an implement; an implement is a tool).

After more than a decade's work toward automatically adapting machine-readable dictionaries for NLP, the field's overall conclusion (Ide & Véronis, 1993) was that this line of research had little direct utility: machine-readable dictionaries simply required too much human-level interpretation to be of much use to machines.

However, traditional dictionaries do not exhaust the available human-oriented lexical resources. The lexical knowledge base called WordNet (Miller, 1995) attempts to record not only what a person knows about words and phrases but also how that knowledge might be organized in the human mind, guided by insights from cognitive science. Begun in the 1980s by George Miller at Princeton University's Cognitive Science Laboratory, the English WordNet project has developed a lexical database organized as a semantic network of four directed acyclic graphs, one for each of the major parts of speech: noun, verb, adjective,

and adverb. Words are grouped into sets of cognitive synonyms, called synsets. Synsets within a part-of-speech network are connected by a small number of relations. For nouns, the main ones are subsumption ("is a") and meronymy ("has as part": *hand has-as-part finger*); for adjectives, antonymy; and for verbs, troponymy (indication of manner: *whisper troponym-of talk*). WordNet itself offers few relations across parts of speech, although satellite projects have pursued aspects of this knowledge gap.

WordNet was adopted by the NLP community for a similar reason as machine-readable dictionaries were: it was large and available. Moreover, its hierarchical structure captured additional aspects of lexical and ontological knowledge that had promise for machine reasoning in NLP. However, WordNet has proved suboptimal for NLP for the same reasons as machine-readable dictionaries did: the ambiguity arising from polysemy. For example, at time of writing *heart* has ten senses in WordNet: two involve a body part (working muscle; muscle of dead animal used as food); four involve feelings (the locus of feelings; courage; an inclination; a positive feeling of liking); two involve centrality (physical; nonphysical); one indicates a drawing of a heart-shaped figure; and one is a playing card. For human readers, the full definitions, synonyms, and examples make the classification clear, but for machines they introduce additional ambiguity. For example, the synonym for the "locus of feelings" sense is "bosom," which has eight of its own WordNet senses. So, although the lexicographical quality of this manually acquired resource is high, interpreting the resource without human-level knowledge of English can be overwhelming.

The consequences of polysemy became clear when WordNet was used for query expansion in knowledge retrieval applications. Query expansion is the reformulation of a search term using synonymous key words or different grammatical constructions. But, as reported in Gonzalo et al. (1998), success has been limited because badly targeted expansion—using synonyms of the wrong meaning of a keyword—degrades performance to levels below those when queries undergo no expansion at all. A relevant comparison is the utility of a traditional monolingual thesaurus to native speakers versus its opaqueness to language learners: whereas native speakers use a thesaurus to jog their memory of words whose meanings and usage contexts they already know, language learners require all of those distinguishing semantic and usage nuances to be made explicit.

Various efforts have been launched toward making the content of WordNet better suited to NLP. For example, select components of some definitions have been manually linked to their correct WordNet senses as a method of disambiguation, and some cross-part-of-speech relations have been added, as between nouns and verbs. Much effort has also been devoted to developing multilingual wordnets and bootstrapping wordnets from one language to another. In the context of this flurry of development, what has not been pursued is a community-wide assessment of whether wordnets, in principle, are the best target of the NLP community's resource-building efforts.

1.6.8 Coreference in the Knowledge-Lean Paradigm

The complexity of reference resolution—of which establishing textual *co*references is just one aspect—has been inadvertently masked by the selective nature of mainstream work in NLP over the past twenty-five years. The vast majority of that work has applied machine learning (most often, supervised) to the simpler instances of the simpler types of referring expressions. To give just a few examples, most systems exclude ellipsis wholesale, they treat pronouns only in contexts in which their antecedents are realized as a single NP constituent, they consider only identity relations, and they consider the identification of a textual coreferent the end point of the task. (Why these constitute only partials is explained in chapter 5.) This means, for example, that *they* in (1.3) will be outside of purview, even though it is far from a worst case as real-world examples go.[37]

(1.3) My dad served with a Mormon and they became great friends. (COCA)

The rule-in/rule-out conditions are encoded in the corpus annotation guidelines that support the machine learning.[38]

An example of a task specification that has significantly influenced work on reference in NLP for the past two decades is the MUC-7 Coreference Task (Hirschman & Chinchor, 1997). This task was formulated to support a field-wide competition among NLP systems. Since it provided developers with annotated corpora for both the training and the evaluation stages of system development, it strongly encouraged the methodology of supervised machine learning. As regards the task's purview, the selection of so-called markables (entities for which systems were responsible) was more strongly influenced by practical considerations than scientific ones. For example, two of the four requirements were the need for greater than 95% interannotator agreement and the ability of annotators to annotate quickly and therefore cheaply—which necessitated the exclusion of all complex phenomena. The other two requirements involved supporting the MUC information extraction tasks and creating a useful research corpus outside of the MUC extraction tasks. Mitkov (2001) and Stoyanov et al. (2009) present thoughtful analyses of the extent to which such simplifications of the problem space have boosted the popular belief that the state of the art is more advanced that it actually is. Stoyanov et al. write, "The assumptions adopted in some evaluations dramatically simplify the resolution task, rendering it an unrealistic surrogate for the original problem." In short, task specifications of this sort—which have been created for quite a number of linguistic phenomena beside coreference—can be useful in revving up enthusiasm via competitions and fostering work on machine learning methods themselves. However, there is an unavoidable negative consequence of removing all difficult cases a priori: few people reading about the results of such systems will understand that the evaluation scores reflect performance on the easier examples. Tactically speaking, this makes it difficult to make the case that much more work is needed on reference—after all, numbers like 90% precision stick in the mind, no matter what they actually mean.

To reiterate, most of the NLP-oriented reference literature over the past twenty-five years has reported competing paradigms of machine learning, along with supporting corpus annotation efforts and evaluation metrics. Olsson (2004) and Lu and Ng (2018) offer good surveys. Poesio, Stuckardt, and Versley (2016; hereafter, PS&V) provide a more comprehensive overview of the field to date. Not only does this collection nicely frame the reference-oriented work described here, the authors also give a mainstream-insider's analysis of the state of the art that, notably, resonates with our own, out-of-the-mainstream observations. In their concluding chapter, "Challenges and Directions of Further Research," PS&V juxtapose the noteworthy advances in reference-related *engineering* with the state of treating *content*:

> If, however, one looks at the discipline from the side of the phenomenon (i.e. language, discourse structure, and—ultimately—*content*), we might arrive at the somewhat sobering intermediate conclusion that, after more than four decades of research, we are yet far away from the ambitious discourse processing proposals propagated by the classical theoretical work. That is, instead of investigating the celestial realms of rhetorical and thematic structure, we're yet occupied with rather mundane issues such as advanced string matching heuristics for common and proper nouns, or appropriate lexical resources for elementary strategies, e.g., number-gender matching etc. (p. 488)

They suggest that we might need to become "more ambitious again" (p. 488) in order to enhance the current levels of system performance. Although we wholeheartedly agree with the spirit of this assessment, we see a danger in describing rhetorical and thematic structure as "celestial realms," as this might suggest that they are permanently out of reach. Perhaps a more apt (and realistic) metaphor would have them on a very tall mountain.

It is noteworthy that PS&V are not alone in their assessment that the field has a long way to go—or, as Poesio puts it: "Basically, we know how to handle the simplest cases of anaphoric reference/coreference, anything beyond that is a challenge." (PS&V, pp. 490–491). For example, among the respondents to their survey about the future of the field was Marta Recasens, who wrote:

> I think that research on coreference resolution has stagnated. It is very hard to beat the baseline these days, state-of-the-art coreference outputs are far from perfect, and conferences receive less and less submissions on coreference. What's the problem? The community has managed to do our best with the "cheapest" and simplest features (e.g., string matching, gender agreement), plus a few more sophisticated semantic features, and this is enough to cover about 60% of the coreference relations that occur in a document like a news article, but successfully resolving the relations that are left requires a rich discourse model that is workable so that inferences at different levels can be carried out. This is a problem hindering research not only on coreference resolution but many other NLP tasks. (PS&V, p. 498)

Although we enthusiastically incorporate, as heuristic evidence, the results of a knowledge-lean coreference resolution engine into our NLU process, this paradigm of work does not inform our own research. Instead, our research is focused on semantically vetting—and, if needed, overturning—the results of such systems, as well as treating the more difficult phenomena that, to date, have been outside of purview. The reasons why the knowledge-lean paradigm does not inform our work are as follows:

1. It does not involve cognitive modeling, integration into agent systems, or the threading of reference resolution with semantic analysis.
2. The results are not explanatory.
3. Many contributions focus on a single reference phenomenon rather than seeking generalizations across phenomena.
4. The work does not involve linguistically grounded microtheories that can be improved over time in service of ever more sophisticated LEIAs. Instead, in the knowledge-lean paradigm, once the machine-learning methods have exploited the available corpus annotations, the work stops, with developers waiting for more and better annotations.

In fact, in response to the same survey mentioned above, Roland Stuckardt noted a complication of the supervised machine learning paradigm in terms of annotation and evaluation:

> The more elaborated the considered referential relations are, the less clear it becomes what "*human-like performance*" really amounts to. Eventually—since the reference processing task to be accomplished is too "vague" and thus not amenable to a sufficiently exact definition—, we might come to the conclusion that it is difficult to evaluate such systems in isolation, so that we have to move one level upwards and to evaluate their contribution chiefly extrinsically at application level. (PS&V, p. 491)

To sum up, knowledge-lean coreference systems serve our agent system in the same way as knowledge-lean preprocessing and syntactic analysis: all of these provide heuristic evidence that contributes to the agent's overall reasoning about language inputs.

1.6.9 Dialog Act Detection

The flow of human interaction overall, and language use in particular, follows typical patterns.[39] For example, upon meeting, people usually greet each other; a question is usually followed by an answer; and a request or order anticipates a response promising compliance or noncompliance. Of course, there are many variations on the theme, but those, too, are largely predictable: for example, the response to a question could be a clarification question or a comment about its (ir)relevance. In agent systems, understanding dialog acts[40] like these is a part of overall semantic/pragmatic analysis.

Automatic dialog act detection using supervised machine learning has been pursued widely enough to be the subject of survey analyses, such as the one in Král and Cerisara

(2010), which covers both the challenges of the enterprise and the methods that have been brought to bear. Among the challenges is creating a taxonomy of dialog acts that, on the one hand, balances the utility of a domain-neutral approach with the necessity for application-specific modifications and, on the other hand, supports an annotation scheme that is simple and clear enough to permit good interannotator agreement. Methods that have been brought to bear include various machine learning algorithms that use features categorized as lexical (the words used in an utterance), syntactic (word ordering and cue phrases), semantic (which can be quite varied in nature, from general domain indicators to frame-based interpretations of expected types of utterances), prosodic, and contextual (typically defined as the dialog history, with the previous utterance type being most important). Král and Cerisara note that application-independent dialog act–detection systems often use all of the above except semantic features.

Traum (2000) attends to the deep-semantic/discourse features that would be needed to fully model the dialog act domain. For example, since speaker intention is a salient feature of dialog acts, mindreading must be modeled; since user understanding is a salient feature, interspeaker grounding must be modeled; and since dialog acts belong to and are affected by the context (defined as the interlocutors' mental models), context must be modeled.

One noteworthy problem in comparing taxonomies of dialog acts is the use of terminology. In narrow-domain applications, the term *dialog act* can be used for what many would consider events in domain scripts. For example, in Jeong and Lee's (2006) flight reservation application, "Show Flight" is considered a dialog act, whereas under a more domain-neutral approach, the dialog act might be *request-information*, with the semantic content of the request being treated separately.

For illustration, we will consider the dialog act inventory in Stolcke et al. (2000),[41] which we selected for two reasons: first, because it includes a combination of generic and application-specific elements; and second, because the selections are justified by their utility in serving a particular goal—in this case, improving a speech recognition system. The latter reminds us of an important facet of statistical approaches: the right features are the ones that work best.

Stolcke et al.'s (2000) inventory of forty-two dialog acts was seeded by the Dialogue Act Markup in Several Layers (DAMSL) tag set (Core & Allen, 1997) and then modified to suit the specificities of their corpus: the dialogs in the Switchboard corpus of human-human conversational telephone speech (Godfrey et al., 1992). Although Stolcke et al. present the speech acts as a flat inventory (p. 341), we classify them into four categories to support our observations about them.[42]

- *Assertions*: STATEMENT, OPINION, APPRECIATION, HEDGE, SUMMARIZE/REFORMULATE, REPEAT-PHRASE, HOLD BEFORE ANSWER/AGREEMENT, 3RD-PARTY-TALK, OFFERS, OPTIONS & COMMITS, SELF-TALK, DOWNPLAYER, APOLOGY, THANKING

- *Question types*: YES-NO-QUESTION, DECLARATIVE YES-NO-QUESTION, WH-QUESTION, DECLARATIVE WH-QUESTION, BACKCHANNEL-QUESTION, OPEN-QUESTION, RHETORICAL-QUESTIONS, TAG-QUESTION
- *Responses*: YES ANSWERS, AFFIRMATIVE NON-YES ANSWERS, NO ANSWERS, NEGATIVE NON-NO ANSWERS, REJECT, RESPONSE ACKNOWLEDGMENT, AGREEMENT/ACCEPT, MAYBE/ACCEPT-PART, DISPREFERRED ANSWERS, BACKCHANNEL/ACKNOWLEDGE, SIGNAL NON-UNDERSTANDING, OTHER ANSWERS
- *Other*: ABANDONED/UNINTERPRETABLE, CONVENTIONAL-OPENING, CONVENTIONAL CLOSING, QUOTATION, COLLABORATIVE COMPLETION, OR-CLAUSE, ACTION-DIRECTIVE, NON-VERBAL, OTHER

If one looks at this inventory in isolation—that is, from a linguistic perspective, divorced from a machine learning application—questions naturally come to mind. Why the fine-grained splitting of question types? Why are APPRECIATE, APOLOGY, and THANKING included while other types of performative acts are excluded? Why is QUOTATION separate from the content of the quotation? However, when the inventory is framed within its intended task, it makes much more sense. Stolcke et al. (2000) write that they "decided to label categories that seemed both inherently interesting linguistically and that could be identified reliably. Also, the focus on conversational speech recognition led to a certain bias toward categories that were lexically or syntactically distinct (recognition accuracy is traditionally measured including all lexical elements in an utterance)" (p. 343).

We appreciate Stolcke et al.'s (2000) clarity of presentation, not only with respect to their goals and experimental results but also with respect to a simplification that boosted their evaluation score. Namely, they provided their system with correct utterance-level segmentations as input, since computing utterance-level segmentations is a difficult and error-prone task in itself. They explain that different developer choices make it difficult to compare systems: "It is generally not possible to directly compare quantitative results because of vast differences in methodology, tag set, type and amount of training data, and, principally, assumptions made about what information is available for 'free' (e.g., hand-transcribed versus automatically recognized words, or segmented versus unsegmented utterances)" (p. 363). This is a good reminder to us all of how essential it is to *read* the literature rather than skim the tables of results.

1.6.10 Grounding

The term *grounding* has been used with various meanings in AI. The two meanings most salient for robotic systems are *linking words to their real-world referents* and *linking any perceptual inputs to agent memory*. We will discuss those in chapter 8.[43] Here, by contrast, we focus on the meaning of *grounding* that involves overtly establishing that the

speaker and interlocutor have achieved mutual understanding, which is a natural and necessary part of a fluid dialog. In live interactions, grounding is carried out through a combination of body language (e.g., maintaining appropriate eye contact and nodding) and utterances (e.g., "hmmm," "uh huh," and "yeah"). In computer dialog systems, by contrast, language is the only available channel for grounding.

Clark and Schaefer (1989, p. 262) posit the *grounding criterion*: "The contributor and the partners mutually believe that the partners have understood what the contributor meant to a criterion sufficient for current purposes." Traum (1999a, p. 130) divides this into two features: how much grounding is enough and how important it is for this level of grounding to be achieved. Baker et al. (1999) focus on the collaborative nature of grounding and the relevance of Clark and Wilkes-Gibbs' (1986) *principle of least collaborative effort*. Baker et al. say that it is better for addressees to simply show that they are listening rather than display exactly how they understand each utterance; if common ground is lost, repair should only be undertaken if it is deemed worth the effort.

Although the intuitions underlying grounding are clear, it is a big leap from intuitions to a formal, computable model. Traum (1999a) took this leap, compiling expectations about grounding into a state transition table covering the following grounding acts: *initiate*, *continue*, *acknowledge*, *repair*, *request repair*, *request acknowledgment*, and *cancel*. For example, if the dialog state is "Need for acknowledgment by initiator" and the responder continues talking without providing that acknowledgment, then the dialog remains in an ungrounded state. Although the model is compellingly formal, Traum himself points out its outstanding needs: the binary grounded/ungrounded distinction is too coarse; typical grounding practices (e.g., how often grounding is expected and needed) differ across language genres and contexts; the automatic identification of utterance units is an unsolved problem, as is the identification of which grounding act was performed (i.e., vagueness and partial understanding/grounding are typical outcomes that would need to be handled by an enhanced model). Traum asks a good question: "While it is clear that effective collaborative systems must employ the use of grounding-related feedback, what is less clear is whether there must be an explicit model of grounding that is referred to in the system's performance and interpretation of communications, or whether a system could be designed to behave properly without such an explicit model." He suggests that his grounding model could be improved by incorporating things like the cost and utility of grounding in conjunction with various other considerations, such as the utility of other actions that could help to ground the utterance.

We are not aware of any substantial breakthroughs in operationalizing models of grounding, which is not surprising since the difficult problems that Traum (1999a) indicates—as well as others he does not, such as the full semantic analysis needed to detect grounding-related features—remain open research issues.

1.6.11 More on Empirical NLP

In its purest form, empirical NLP relies on advanced statistical techniques for measuring similarities and differences between textual elements over large monolingual or multilingual text corpora—with corpora being viewed as repositories of evidence of human language behavior. In corpus-based approaches, all feature values must be obtained from unadorned text corpora.[44] That is, the only knowledge that exists is the surface form of text, as we would read it online or in a book. Within this neobehaviorist paradigm, there is no need to overtly address unobservables such as meaning; in fact, the very definition of *meaning* shifted. For example, in the latent semantic analysis approach, word meaning is understood essentially as a list of words that frequently appear in texts within N words of the "target" word whose meaning is being described. By the time of this writing, the empiricist paradigm in NLP has matured, and its main issues, results, and methods are well presented in the literature (for overviews, see, e.g., Jurafsky & Martin, 2009; Manning & Schütze, 1999).

One hallmark of recent NLP has been a widespread preference for developing—often in the context of a field-wide competition[45]—component technologies over building end-user applications. This preference has usually been justified as learning to walk before learning to run, or, in a more scholarly fashion, by saying that the scientific method mandates meeting prerequisites for a theory or a model before addressing that theory or model as a whole. In fact, in NLP, the latter precept has been often honored in the breach: in many (perhaps most?) cases, theoretical work on a variety of language phenomena proceeds from the assumption that all the prerequisites for the theory are met, whereas in reality this is seldom the case. This exasperates developers of application systems on the lookout for readily available, off-the-shelf components and knowledge resources for boosting the output quality of their applications. Their appetites are whetted when they read the description of a theory that promises to help them solve a practical problem, only to realize on further investigation that the theory can work only if certain currently unattainable prerequisites are met. For example, if a theory claims to solve the problem of automatically determining the discourse focus in a dialog but requires a complete propositional semantic analysis of the dialog content as a prerequisite, then it will not be of any use to practical dialog system builders because full semantic analysis is currently beyond the state of the art. It is in this context that one must understand the famous quip by Fred Jelinek, a leader in the field of automatic speech recognition, to the effect that every time he fired a linguist, his system's results improved.

Here we consider just two examples of tasks whose results are not directly useful for NLU because the task specification itself contrasts too markedly with the goals of full NLU. The tasks in question are word sense disambiguation and the interpretation of nominal compounds.

Word sense disambiguation. Within the empiricist paradigm, word sense disambiguation (WSD) has been identified as a freestanding task, which has been approached using both

supervised and unsupervised machine learning. Associated with each approach is, interestingly enough, a different goal (see Navigli's 2009 survey for details). WSD using supervised machine learning is a classification task: the system is required to assign instances of words to a closed set of word meanings (selected by task developers) after training on an annotated corpus that provides word-to-meaning correspondences. In targeted WSD, systems are expected to disambiguate only certain target words, typically one to a sentence, for which ample training evidence (annotated examples) is provided. In all-words WSD, systems are expected to disambiguate all open-class words, but data sparseness (i.e., lack of sufficient training examples for each word) impedes the quality of results. By contrast, WSD using unsupervised machine learning is a clustering task whose goal is to cluster examples that use the same sense of a word. Although motivations for pursuing WSD as an independent task have been put forth (see, e.g., Wilks, 2000), when seen from an agent-building perspective, this is incongruent, since the results of WSD become ultimately useful only when they are integrated with dependency determination, reference resolution, and much more.

Identifying the relations in nominal compounds. Nominal compounding has been studied by descriptive linguists, psycholinguists, and practitioners of NLP.[46] Descriptive linguists have primarily investigated the inventory of relations that can hold between the component nouns. They have posited anywhere from six to sixty or even more descriptive relations, depending on their take on an appropriate grain size of semantic analysis. They do not pursue algorithms for disambiguating the component nouns, presumably because the primary consumers of linguistic descriptions are people who carry out such disambiguation automatically. However, they do pay well-deserved attention to the fact that NN interpretation requires a discourse context, as illustrated by Downing's (1977) "apple-juice seat" example. Psycholinguists, for their part, have found that the speed of NN processing increases if one of the component nouns occurs in the immediately preceding context (Gagné & Spalding, 2006). As for mainstream NLP practitioners, they typically select a medium-sized subset of relations of interest and train their systems to automatically choose the relevant relation during the analysis of compounds taken outside of context—that is, presented as a list. Two methods have been used to create the inventory of relations: developer introspection, often with iterative refinement (e.g., Moldovan et al., 2004), and crowdsourcing, also with iterative refinement (e.g., Tratz & Hovy, 2010). A recent direction of development involves using paraphrases as a proxy for semantic analysis: that is, a paraphrase of an NN that contains a preposition or a verb is treated as the meaning of that NN (e.g., Kim & Nakov, 2011). However, since verbs and prepositions are also highly ambiguous, these paraphrases do not count as fundamental disambiguation. Evaluations of knowledge-lean systems typically compare machine performance with human performance on a relation-selection or paraphrasing task.

In most statistical NLP systems, the semantics of the component nominals is not directly addressed: that is, semantic relations are used to link uninterpreted nouns. Although this is incongruous from a linguistic perspective, there are practical motivations.

1. The developers' purview can be a narrow, technical domain (e.g., medicine, as in Rosario & Hearst, 2001) that includes largely monosemous nouns, making nominal disambiguation not a central problem.[47]
2. The development effort can be squarely application-oriented, with success being defined as near-term improvement to an end system, with no requirement that all aspects of NN analysis be addressed.
3. The work can be method-driven, meaning that its goal is to improve our understanding of a machine learning approach itself, with the NN dataset being of secondary importance.
4. Systems can be built to participate in a field-wide competition, for which the rules of the game are posited externally (cf. the Free Paraphrases of Noun Compounds task of SemEval-2013 in Hendrickx et al., 2013).

Understanding this broad range of developer goals helps not only to put past work into perspective but also to explain why the full semantic analysis approach we will describe in chapter 4 does not represent an evolutionary extension of what came before; instead, it addresses a different problem altogether. It is closest in spirit to the work of Moldovan et al. (2004), who also undertake nominal disambiguation. However, whereas they implement a pipeline from word sense disambiguation to relation selection, we combine these aspects of analysis.

1.6.12 Manual Corpus Annotation: Its Contributions, Complexities, and Limitations

Corpus annotation has been in great demand over the past three decades because manually annotated corpora are the lifeline of NLP based on supervised or semisupervised machine learning (Ide & Pustejovsky, 2017). However, despite the extensive effort and resources expended on corpus annotation, the annotation of meaning has not yet been addressed to a degree sufficient for supporting NLP in the framework of cognitive modeling. So, even though annotated corpora represent a gold standard, the question is, What is the *gold* in the standard? The value of the gold derives from the task definition for the annotation effort, which in turn derives from developers' judgments about practicality and utility. To date, these judgments have led to creating annotated corpora to support such tasks as syntactic parsing, establishing textual coreference links, detecting proper names, and calculating light-semantic features, such as the case role fillers of verbs. Widely used annotated corpora of English include the syntax-oriented Penn Treebank (e.g., Taylor et al., 2003); PropBank, which adds semantic role labels to the Penn Treebank (Palmer et al., 2005); the Automatic Content Extraction (ACE) corpus, which annotates semantic relations and events (e.g., Doddington et al., 2004); and corpora containing annotations of pragmatics-oriented phenomena, such as coreference (e.g., Poesio, 2004), temporal relations (e.g., Pustejovsky et al., 2005), and opinions (e.g., Wiebe et al., 2005).

Decision-making about the scope of phenomena to annotate has typically been more strongly affected by judgments of practicality than utility. Some examples:

- The goal of the Interlingual Annotation of Multilingual Text Corpora project (Dorr et al., 2010) was to create an annotation representation methodology and test it on six languages, with component phenomena restricted to those aspects of syntax and semantics that developers believed could be consistently handled well by the annotators for all languages.
- When extending the syntactically oriented Penn Treebank into the semantically supplemented PropBank, developers selected semantic features (coreference and predicate argument structure) on the basis of feasibility of annotation (Kingsbury & Palmer, 2002).
- The scope of reference phenomena covered by the MUC coreference corpus was narrowly constrained due to the requirements that the annotation guidelines allow annotators to achieve 95% interannotator agreement and to annotate quickly and, therefore, cheaply (Hirschman & Chinchor, 1997).

Before passing an opinion about whether annotation efforts have been sufficiently ambitious, readers should pore over the annotation guidelines compiled for any of the past efforts, which grow exponentially as developers try to cover the overwhelming complexity of real language as used by real people. As Sampson (2003) notes in his thoughtful review of the history of annotation efforts, the annotation scheme needed to cover the syntactic phenomena in his corpus ran to 500 pages—which he likens both in content and in length to the independently produced 300+ page guidelines for Penn Treebank II (Bies et al., 1995). Hundreds of pages for syntax alone—we can only imagine what would be needed to cover semantics and discourse as well.

Since interannotator agreement and cost are among the most important factors in annotation projects, semiautomation—that is, automatically generating annotations to be checked and corrected by people—has been pursued in earnest. Marcus et al. (1993) report an experiment revealing that semiautomating the annotation of parts of speech and light syntax in English doubled annotation speed, showed about twice as good interannotator agreement, and was much less error-prone than manual tagging. However, even though semiautomation can speed up and improve annotation for simpler tasks, the cost should still not be underestimated. Brants (2000) reports that although the semiautomated annotation of German parts of speech and syntax required approximately fifty seconds per sentence, with sentences averaging 17.5 tokens, the actual cost—counting annotator training and the time for two annotators to carry out the task, for their results to be compared, and for difficult issues to be resolved—added up to ten minutes per sentence.

The cost of training and the steepness of the training curve for annotation cannot be overstated. Consider just a few of the rules comprising the MUC-7 task definition (Chinchor, 1997) for the annotation of named entities. Family names like *the Kennedys* are not

to be annotated, nor are diseases, prizes, and the like named after people: *Alzheimer's*, *the Nobel prize*. Titles like *Mr.* and *President* are not to be annotated as part of the name, but appositives like *Jr.* and *III* ("the third") are. For place names, compound place names like *Moscow, Russia* are to be annotated as separate entities, and adjectival forms of locations are not to be annotated at all: *American companies*. While there is nothing wrong with these or any comparable decisions about scope and strategy, lists of such rules are very hard to remember—and one must bear in mind that tagging named entities, in the big picture of text annotation, is one of the simplest tasks.

This leads us to a seldom discussed but, in our opinion, central aspect of corpus annotation: it is expensive and labor-intensive, not to mention unpleasant and thankless—a combination of factors that puts most actual annotation work in the hands of low-paid students.

The empirical, machine learning–oriented paradigm of NLP has been routinely claimed to be the realistic alternative to knowledge-based methods that rely on expensive knowledge acquisition, but corpus annotation *is* expensive knowledge acquisition. The glamorous side of the work in this paradigm is the development and evaluation of the stochastic algorithms that use these annotations as input.

It is possible that during the early stages of the neobehaviorist revival, the crucial role of training materials for learning how to make sophisticated judgments by analogy was not fully appreciated. But unsupervised learning, although the cleanest theoretical concept, has so far proved to be far less successful. The preconditions of supervised learning put the task of corpus annotation, and the concomitant expense, front and center. The little-acknowledged reality is that the complexity and extent of the annotation task is fully commensurate with the task of acquiring knowledge resources for knowledge-based NLU. One lesson to learn from this is that the need for knowledge simply does not go away with a change in processing paradigms. And one thing to remember about corpus annotations is that, in contrast to knowledge bases developed for NLP, there is a big leap from examples to the kinds of useful generalizations that machine learning is expected to draw from them.

Although most annotation efforts to date have focused on relatively simpler phenomena, not all have. For example, the Prague Dependency Treebank (PDT) is a complex, linguistically motivated treebank that captures the deep syntactic structure of sentences (Mikulová, 2014). It follows a dependency-syntax theory called Functional Generative Description, according to which sentences are represented using treelike structures comprised of three interlinked layers of representation: the morphological layer, the surface syntactic (analytical) layer, and the deep syntactic (tectogrammatical) layer. The latter captures "the deep, semantico-syntactic structure, the functions of its parts, the 'deep' grammatical information, coreference and topic-focus articulation including the deep word order" (Mikulová, p. 129). The representations include three vertically juxtaposed and interlinked tree structures. Among the PDT's noteworthy features is its annotation of two types of deletions: textual ellipsis, in which the deleted material could have been expressed in the surface syntax (even if this would have led to stylistic infelicity), and grammaticalized

ellipsis, in which some meaning must be semantically reconstructed but no corresponding category could be inserted into the surface syntax (Hajič et al., 2015). Deletions are accounted for in the PDT by introducing nodes in the tectogrammatical layer. Since Czech is a subject-drop language, this node-introduction strategy is widely represented in the PDT. However, introducing nodes is not the only way that null subjects have been treated in annotation schemes. According to Hajič et al., the treebanks of Italian and Portuguese—not to mention the analytical layer of the PDT—do not include such nodes.

The literature describing the PDT illustrates just how much theoretical and descriptive work must underpin the development of an annotation scheme before annotators are even set to the practical task. For example, Marie Mikulová et al.'s "Annotation on the Tectogrammatical Layer in the Prague Dependency Treebank"[48] runs to over 1,200 pages—a size and grain size of description that rivals comprehensive grammars. Similarly, a book-length manuscript (Mikulová, 2011) is devoted entirely to the identification and representation of ellipsis, without even opening up issues related to conditions of usage, their explanations, or predictive heuristics.

In the early twenty-first century, corpus annotation—specifically, creating the theoretically grounded annotation guidelines—has been the most visible arena for descriptive linguists to flex their muscles. The purview of descriptive linguistics has expanded from idealized, well-behaved, most-typical realizations of phenomena to what people actually say and write. The corpora annotated using such schemes can serve further linguistic investigation by making examples of phenomena of interest identifiable using simple search functions. In fact, Hajič et al. (2016, p. 70) present an in-depth analysis of how the process of annotating the PDT, as well as its results, have led to amendments in the underlying linguistic theory and a better understanding of the language system.

1.7 Further Exploration

1. There are many hard things about language. One of them is understanding bad writing. Read or watch Steven Pinker's insightful and entertaining analyses of bad writing and its good counterpart:

- *The Sense of Style: The Thinking Person's Guide to Writing in the 21st Century* (Penguin, 2014).
- "Why Academics Stink at Writing," *The Chronicle Review*, The Chronicle of Higher Education, September 26, 2014, https://stevenpinker.com/files/pinker/files/why_academics_stink_at_writing.pdf
- Various lectures available on YouTube, such as "Linguistics, Style and Writing in the 21st Century—with Steven Pinker," October 28, 2015, https://www.youtube.com/watch?v=OV5J6BfToSw&t=1020s

2. The history of machine translation makes for interesting reading. Some suggestions:

- Warren Weaver's 1949 memorandum "Translation," available at http://www.mt-archive.info/Weaver-1949.pdf
- Yehoshua Bar Hillel's (Hebrew University, Jerusalem) "The Present Status of Automatic Translation of Languages," from *Advances in Computers*, vol. 1 (1960), pp. 91–163, available at http://www.mt-archive.info/Bar-Hillel-1960.pdf
- John Hutchins's "ALPAC: The (In)famous Report," available at http://www.hutchinsweb.me.uk/ALPAC-1996.pdf
- *Readings in Machine Translation*, edited by S. Nirenburg, H. Somers, and Y. Wilks (MIT Press, 2003), which contains all of the above as well as many other relevant texts.

3. Investigate the current state of the art in machine translation using Google Translate (translate.google.com). You don't need to know another language to do this.

- Copy-paste (or simply type) a passage into the left-hand window and be sure it is recognized as English.
- Translate it into any of the available languages by choosing a target language in the right-hand window.
- Copy the translation (even though you won't understand it) back into the left-hand window and be sure the system understands which language it is.
- Translate the translation back into English.

  a. How good is the translation?

  b. Can you hypothesize any differences between English and that language based on the output? For example, maybe that language does not use copular verbs (i.e., the verb *be* in sentences like *George is a zookeeper*), or maybe it permits subject ellipsis—both of which might be reflected in the translation back into English.

You should get better translations if you (a) select a language, L, for which L-to-English and English-to-L machine translation has been worked on extensively (e.g., French, Spanish, Russian); (b) select a language that is grammatically close to English; and (c) select a grammatically normative text (not, e.g., a highly elliptical dialog). Make the opposite choices and translation quality is likely to suffer. If you know another language, things become more interesting since you can do multistage translation—not unlike the telephone game, in which players whisper a message in a circle and see how much it morphs by the time it reaches the last player.

4. Read about the mainstream approaches to NLP over the past thirty years in Jurafsky and Martin's *Speech and Language Processing: An Introduction to Natural Language*

Processing, Speech Recognition, and Computational Linguistics, 2nd ed. (Prentice-Hall, 2009).

5. Think about and/or discuss the differences between applications that operate over big data (e.g., question-answering *Jeopardy!*-style) and applications in which every utterance is produced exactly once, using exactly one formulation (e.g., a task-oriented dialog). What are the challenges and opportunities specific to each one?