Proceedings of the Second Annual Conference on Advances in Cognitive Systems ACS-2013 (39-56)

Understanding Social Interactions Using Incremental Abductive Inference

Ben Meadows BMEA011@AUCKLANDUNI.AC.NZ
Pat Langley PATRICK.W.LANGLEY@GMAIL.COM
Miranda Emery MEME011@AUCKLANDUNI.AC.NZ
Department of Computer Science, University of Auckland, Private Bag 92019, Auckland 1142, NZ

Abstract

In this paper we present a computational approach to key aspects of understanding social interactions. First, we specify a class of problems – understanding fables – that require inference about agents' mental states from their behavior. After this, we review earlier work on UMBRA, an abductive system for single-agent plan understanding, and describe extensions that let it deal with multi-agent scenarios, including ones that involve accidental errors and intentional deceptions. These augmentations include distinguishing domain-level knowledge from more general content about social interactions and applying this knowledge at nested levels of belief. We also report results on a set of fable-based scenarios that demonstrate the benefits of these extensions. In closing, we discuss how our approach to social cognition is informed by earlier research in the area.

"Oh, what a tangled web we weave / When first we practice to deceive!"
– Sir Walter Scott (*Marmion, 1808)*

1. Introduction

Understanding the nuances of social interactions is a sufficiently routine task that people usually do it without conscious effort. However, *routine* is not the same as *simple*. In social settings, we continuously generate hypotheses about others' beliefs, about their goals when performing social actions, and about whether reality matches their beliefs or satisfies their goals. In some cases, explaining social behavior requires the ability to infer others' ignorance of the true situation and even their intent to deceive third parties. Moreover, it can involve thinking about how agents reason about others and how they reflect upon their own social interactions. All of these facets of social cognition rely on constructing models of others' mental states.

Such abilities are a distinctive characteristic of human intelligence, and the aim of our research is to develop a computational account of the representations and processes that underpin them. Such a theory should specify both the stable long-term knowledge and the dynamic short-term elements that support social understanding, and it should also state the mechanisms that apply this knowledge to explain behavior in terms of agents' inferred mental states. Our account should generalize beyond specific physical domains and support reasoning about situations that involve errors and deception.

Before proceeding further, we should note some research paradigms that touch on related topics but differ from our own in important ways. These include:

- *Activity recognition* (e.g., Aggarwal & Ryoo, 2011), an area of AI that deals with classifying an agent's observed behavior into some known category; this differs from our focus because it does not deal with the mental states underlying behavior.
- *Plan recognition* (e.g., Goldman, Geib, & Miller, 1999), another AI subfield that is concerned with inferring the goal or goals that produce observed behavior; this comes closer to our emphasis but typically deals with single agents and focuses on domain-level knowledge.
- *Behavior explanation* (e.g., Malle, 1999), a movement in social psychology that studies inferences about the causes, reasons, and intentions behind human actions; its aims are close to our own, but the paradigm has not yet produced computational models.
- *Collaborative planning* (e.g., Rao, Georgeff, & Sonenberg, 1992; Bond, 2002), an area of AI that deals with forming joint plans among multiple agents with shared goals; work in this tradition often encodes other agents' beliefs and goals (e.g., Castelfranchi, 1998), but focuses on generation rather than understanding.
- *Story understanding* (e.g., Wilensky, 1978; Mueller, 2002), a subfield of natural language processing that aims to interpret and answer questions about stories, some of which deal with interacting agents; this work often deals with inference of their beliefs and goals, but typically in the context of written stories; we abstract away from the source of information about agent behavior, which need not come from natural language.

Although none of these research areas has precisely the same focus as our own, there are enough common elements that we will incorporate a number of their ideas and assumptions.

In the sections that follow, we report a computational account of social understanding that builds on insights from these prior efforts. Readers who are familiar with this earlier work will see that the elements in this account are not novel, but also that, when taken together, they make an important contribution by providing a coherent system-level approach to modeling social understanding. We start by presenting a suitable class of problems, and then review UMBRA, an earlier system for single-agent plan understanding that forms the basis of our response. Next we describe the representational and processing extensions we have made to let UMBRA support social cognition, followed by experimental studies that demonstrate their efficacy and importance. In closing, we discuss specific pieces of related work in more detail, along with our plans for future research.

2. Social Understanding of Fables

Research in cognitive systems always benefits from some class of problems that suggests important directions to explore and challenges to overcome. We have chosen to focus on Aesop-like *fables* to drive our research on social understanding. These typically involve multiple agents who interact in nontrivial ways, yet the tales themselves are succinct and omit irrelevant domain details. This makes them well-suited to demonstrating social cognition, which is largely orthogonal to reasoning about the physical world. Moreover, as Meehan (1977) notes in his early work on story generation, such fables involve "simple stories whose points, or morals, are [. . .] concerned with general lessons, notions from a higher domain." This suggests they will be a useful vehicle for studying cognitive

Table 1. The eight fables comprising our set of test scenarios.

Fable	**Description**
THE HUNGRY CROW	A crow solves a simple problem: feeling hungry, she travels to a barn and acquires grain by opening a jar.
THE SPYING SNAKE	A snake watches and understands what the crow is doing as she solves the same problem as in 'The Hungry Crow'.
THE PROUD LION	A lion is zealously proud of his mane and willing to attack anyone who dares to attempt such finery. The lion, passing by a river, sees his reflection. He jumps in the river to attack the 'other lion'.
THE SHEEP AND WORMS	A sheep watches a crow eating worms. He mistakenly infers that the worms are good to eat, and follows suit, becoming sick as a result.
THE FOX AND THE CROW	A hungry crow in possession of some sour, inedible grapes trades them to an unwitting fox in return for some delicious grain.
THE LURKING EEL	A fox, finding an eel in a pond (and overestimating his swimming skill), decides to seize it and eat it. The eel drowns the fox instead.
THE TRAPPED CROW	A fox has trapped a crow in a jar. The crow pretends to have died of suffocation in order to trick the fox into letting it escape.
THE LION AND THE SHEEP	A lion is too old to hunt animals. The lion announces he is sick. The sheep, believing he is harmless, follows social convention and visits the lion's caves to pay his respects. The lion kills and devours him.

abilities that generalize beyond narrow domains, although we will not focus on the task of drawing moral lessons here.

Rather than taking fables directly from Aesop, we have designed a suite of vignettes in the same style, in some cases amalgamating or re-telling existing fables. Thus, they use animals as characters in settings that (a) are brief, (b) focus on goal-directed behavior, (c) often center on high-level social interactions that include communication, and (d) involve agents who reason about others' mental states. Each fable describes the participating agents' actions, perceptions, and choices. The main differences from standard fables are that we are not concerned with their morals, and we present things from the perspective of a primary agent rather than narrating to an absent audience.

Table 1 presents these fables, ordered by the complexity of reasoning they require for understanding. For instance, the first two stories involve social interaction only through observation, and we include them to establish that our system can reason about agents' nested models of other agents, sometimes called 'mind reading'. These two scenarios follow the same basic plan but have different levels of embedding in their explanatory structure. For example, in The Hungry Crow, the observer infers that the crow believes the jar is not locked; in The Spying Snake, the observer comes to believe that the snake believes that the crow believes the jar is not locked.

The remaining fables in the table are more complex. In both The Proud Lion and The Sheep and Worms, the observer must infer the central agent's mistaken beliefs and why they arise, while also drawing accurate conclusions about the true events. In The Fox and the Crow, the observer sees the crow recognize the fox's false beliefs – incorrect assumptions about trustworthiness and the edibility of food – and capitalize on them for her own ends; something similar happens in The Lurking Eel. In The Trapped Crow and The Lion and the Sheep, the observer sees the central agent go further, intentionally deceiving another agent in order to enact some plan.

We maintain that understanding these vignettes requires a number of capabilities that are central to social cognition. These include encoding and reasoning about agents' models of other agents' mental states, as well as more sophisticated abilities related to representing false beliefs, taking advantages of those beliefs (opportunism), and encouraging such errors for one's own ends (deception). We now turn to a computational framework that supports reasoning about social interactions.

3. A Review of UMBRA

One of our central theoretical claims is that social understanding involves the construction of explanations through a process of abductive inference. The type of 'everyday reasoning' that such abduction provides is appropriate to understanding tasks in real-world domains, where an agent is unlikely to ever have access to complete information. In previous work (Meadows, Langley, & Emery, 2013), we have reported UMBRA, a system that approaches single-agent plan understanding from this perspective. However, before we describe the extensions required to handle the more sophisticated task of social understanding, we review UMBRA's representations and mechanisms.

3.1 Representation in UMBRA

Like many cognitive architectures (Langley, Laird, & Rogers, 2009), UMBRA divides content into a *working memory* and a *long-term memory*. Working memory stores both information arriving from the environment, such as statements about some agent's behavior, and inferences drawn from this external input. This short-term store contains two types of element – beliefs and goals – stated as logical literals. For example, *belief(lion, prey(sheep))* represents the lion's belief that the sheep is a prey animal, and *goal(lion, eat(lion, sheep))* encodes the lion's goal to eat the sheep. Together, these types of elements are sufficient for the task of single-agent plan understanding.

In contrast, long-term memory contains conceptual knowledge and skills that encode generalized knowledge about situations and activities, similar to that found in hierarchical task networks (Nau et al., 2001). Each conceptual rule associates a predicate in the head with a relational situation described in the body. Each skill or method associates a predicate in the head with a set of preconditions, a set of invariants, a set of postconditions, and a set of subtasks. Higher-level predicates are defined in terms of lower-level ones, imposing a hierarchical organization on long-term memory. The same predicate can appear in the head of different conceptual or skill rules, supporting disjunctive and recursive definitions.

For example, one decomposition of the *hunt(Actor, Prey, Loc)* skill involves a pattern of invariants *agent(Actor)* and *not(dead(Actor))*, subtasks *chase_to(Actor, Prey, Loc)* and *kill(Actor, Prey, Loc)*, and postconditions *dead(Prey)* and *at_location(Prey, Loc)*. The subtasks *chase_to(Actor, Prey,*

Loc) and *kill(Actor, Prey, Loc)* have their own decompositions in terms of preconditions, invariants, postconditions, and actions.

Because of the hierarchical structure of knowledge in long-term memory, the explanations generated by the system can be represented as directed graphs whose leaves are concept literals, applied operators, and constraints. Their roots and non-terminal nodes are rule heads, where non-terminals appear as subtasks in other rule applications and roots do not. Each component rule application in an explanation is annotated with provenance information on whether its conditions were assumed, inferred, or retrieved from memory, similarly to the way working memory elements store information about whether they were introduced via assumption, inference, or external input.

3.2 Processing in UMBRA

UMBRA constructs explanations in an effort to understand its observations. The system's theoretical foundations include three primary tenets:

- Explanation generation operates incrementally as new input observations arrive, with later inferences building on earlier ones.
- Reasoning is abductive in character rather than deductive, in that many inference steps use rules to introduce plausible assumptions rather than drawing strict derivations.
- Inference is data driven, usually involving 'bottom-up' chaining from observations and heads of rules, rather than 'top-down' chaining from queries.

The system operates over a series of *input* cycles in which it receives external information that it attempts to explain. Every input cycle comprises zero or more *inference* cycles, each of which involves a single rule application that adds elements to working memory. The latter corresponds roughly to a recognize-act cycle in a production system architecture, except that UMBRA rule applications only add elements to working memory, so that its construction of explanations is monotonic. Within a single inference cycle, the system:

- Identifies each rule R that has a condition or head C unifiable with some element E currently in working memory;
- Generates, for each candidate rule-element pair R–E, a partially instantiated head H that is based on the unification of C with element E;
- Produces candidate rule instances for each partially instantiated rule head H, in each case minimizing the number of assumptions made to complete the body of rule R;
- Ranks these candidate rule instances by an evaluation function that combines arithmetically the average recency R of matched elements, the total number T of assumptions, and the fraction U of observations and inferred heads in the resulting explanation not explained by other rules.
- *Either* extends the explanation by adding inferences from the lowest-cost candidate to memory, *or*, if the cycle has exceeded the allowed number of assumptions, ends the current input cycle.

This sequence of operations incrementally extends the explanation to incorporate ever more observations and, where needed, default assumptions that connect them. The end result is a coherent, connected, hierarchical account of the input in terms of available background knowledge.[1]

1. UMBRA's design borrows ideas from a number of earlier systems, some of which we will discuss in Section 6.

3.3 Recent Revisions to UMBRA

In previous work, we tested UMBRA on plan understanding tasks that involved inferring the beliefs of single agents (Meadows, Langley, & Emery, 2013). Precision and recall scores from experiments in a standard domain were similar to those of earlier systems that operated on the same tasks in a top-down manner, which suggested that our approach to plan understanding had some promise. These results encouraged us to apply our system to tasks that require social understanding, like those described earlier. However, detailed analyses of earlier runs revealed a number of drawbacks to its operation. These led us to revise UMBRA along several fronts that we believed would support a more robust ability to generate plausible and complete explanations.

The original UMBRA had a low processing speed and a high chance of firing rules that would produce false positives. We addressed these issues by revising the system so that it only retrieves rule instances for application if their head or one of their antecedents unifies with an element whose predicate is not too prevalent in the knowledge base (comprising > 1.5% of all rule elements). In runs on plan understanding tasks over the fable domain, this eliminates about 15% of predicates like *actor* and *at_location*, which are very common and thus contain little information.

UMBRA's evaluation function lets it select the rule instance I to apply on each inference cycle, but we found the average recency of I's matched elements to be an ineffective criterion. We have replaced it in the revised UMBRA with the *proportion*, P, of I's antecedents (plus its head) that it would need to assume.[2] Once UMBRA has generated a set of candidate rule instances, it removes any alternative instance I from the set unless either: (a) I is deductively valid; (b) I's subtasks include a nondefault element that does not appear in any other rule instance; (c) I's antecedents include two nondefault elements that do not appear in any other rule instance, and I's subtasks were not all assumed in the course of applying I; or (d) I's subtasks include two nondefault elements that do not appear together in any other rule instance in the explanation. An element is nondefault if it was provided as input or inferred from a rule head. This restriction keeps the system from producing overelaborate explanations that contain elements we would view as false positives.

Informal studies suggested that these revisions to UMBRA improve its ability to construct coherent and plausible explanations, increasing the number of desirable assumptions it makes (thus raising recall) without increasing the chances of undesirable ones (and reducing precision). They appear to reduce UMBRA's sensitivity to a key parameter, the number of default assumptions allowed per rule application. We will not report these empirical results here, as they are unrelated to the paper's focus on social cognition, but the revised system appears to offer an effective approach to generating explanations for the observed behavior of individual agents.

4. Extensions to UMBRA for Social Understanding

As noted above, our initial studies of UMBRA's behavior on abductive understanding of single-agent plans produced encouraging results, but scenarios that involve social explanations of interacting agents, such as those in our fables, introduce new issues that required additional extensions.

2. The specific arithmetic function is $P + T/15 + U$, where T is the number of assumptions and U is the fraction of structural elements not yet explained by other rule instances. Interestingly, we found that with the augmented UMBRA's final design, the influence of the specific choice of function on the system's outputs decreased significantly.

In this section, we describe our modifications to UMBRA for the purpose of social understanding, focusing first on representational changes and then turning to processing augmentations.

4.1 Representational Extensions

Before UMBRA can reason about social interactions, it must first be able to represent them. One important aspect of such exchanges is that they occur over time. As others have found, we were able to avoid explicit temporal encodings in our earlier work because hierarchical plans for single agents are typically constrained enough to suggest correct interpretations without them. Analyses suggest that temporal relations are more important in social plans; for instance, the time at which one agents says something to another can influence later behavior in important ways.

In response, we have extended our notation for beliefs and goals so that each specifies (1) the agent A holding that belief or goal, (2) the content C of that belief or goal, and (3) the start and end times for the structure. The start time denotes when agent A began to believe C or adopted goal C; the end time encodes when A stopped believing C or abandoned it. Unknown times are denoted by Skolem values. For example, *belief(lion, prey(sheep), 06:00, s1)* represents the lion having a belief that the sheep is a prey animal from time 06:00 to some unspecified time, while *goal(lion, healthy(lion), 12:00, 12:30)* encodes the lion's goal, held from 12:00 to 12:30, to be in good health.

In many social settings, we are concerned with the order in which things happen. This is easy to express when we know the start and end times for events, but the possibility of "Skolemized" times suggests a need for more general ways to encode such relations. To this end, the new UMBRA incorporates *constraints*, a third type of meta-level predicate or mental state that lets it specify various types of ordering, identity, and temporal relations on elements of other structures. For example, *constraint(lion, nequal(sheep, s1), 05:30, s2)* represents the lion's constraint, adopted at 05:30, that the sheep is not identical to some Skolem value *s1*, while *constraint(fox, during(s3, s4, 08:00, s5), 05:35, 06:00)* represents the fox's constraint, held between 05:35 and 06:00, that time period from s3 to s4 occurred between 08:00 and s5. Constraints are first-class structures, at the same level as beliefs and goals. Thus, they can appear both as elements in working memory and as components of rules in long-term memory. Both conceptual knowledge and skills involve constraints, with the former typically emphasizing inequalities and the latter often focusing on temporal relations.

As noted earlier, a central requirement for understanding social interactions is encoding an agent's beliefs, goals and constraints about other agents' mental states. To support this ability, we have extended UMBRA's notation to include embedded structures in which the content of one agent's beliefs and goals may be the beliefs and goals of other agents.[3] For example, *belief(crow, goal(lion, eat(lion, sheep, s3, s4), s5, s6), s1, s2)* denotes that the crow believes from time *s1* to *s2* that the lion has the goal *eat(lion, sheep, s3, s4)* from time *s5* to *s6*. Nor is the system limited to two-level structures; it can represent arbitrarily deep embeddings of beliefs and goals, although we have not needed more than four levels in our work to date.

We should note that embedded mental states occur primarily in working memory, with most rules in long-term memory having a single level. This is because they deal primarily with a single

3. Embedded logical literals are not the only plausible representation; we briefly discuss *worlds* and *partitions* in Section 6. We believe that use of these alternative representations would not require changes to any of our core tenets, only to our mechanisms (such as automatic embedded rule generation for knowledge application).

Table 2. Predicates of rules relevant to fables provided to UMBRA to support social inference.

announce_genuine An agent believes some concept, announces it, and another agent adopts the belief as a result.
announce_wrong An agent believes in some concept, announces it, and another agent (who believes the original agent is believing wrongly) does not adopt the belief as a result.
announce_false An agent does not believe some concept, announces it, and another agent adopts the belief as a result.
interpret_as_real, interpret_as_real_agent, interpret_as_real_attributed An agent interprets something it has perceived as a real entity (with agency or other attributes).
interpret_as_image, interpret_as_image_attributed An agent interprets something it has perceived as a false image (perhaps with certain attributes).
become_jealous An agent is proud of an attribute A and experiences envy due to its belief that another agent has A.
judge_not_a_threat An agent believes that another agent is not a threat due to some reason.
pretend_attribute An agent deceives other actors at its location by pretending to have some attribute (e.g., playing dead).
suggest_trade_good_faith An agent believes its possession is good and offers to trade it for something it believes to be good.
suggest_trade_bad_faith An agent believes its possession is bad and offers to trade it for something it believes to be good.

agent's beliefs and goals about the environment, such as the expected effects of executing a hierarchical skill under certain conditions. However, knowledge about social interactions and judgements is an important exception. A primary purpose of such interactions is to alter others' mental states in goal-directed ways, so rules that describe them must characterize not only how others' goals and beliefs change over time, but also how those changes relate to the actor's goals and beliefs. With this in mind, we have provided UMBRA with social knowledge that encodes such relations.

These skills are typically domain independent, in that they can refer to any content in an agent's belief or goal, although a few of those we created refer to predicates that have intermediate levels of generality. These include rules about pretense (e.g., playing dead), trusting other agents' statements, transactions, viewing another agent as a threat, and being jealous of another's property. Nevertheless, each of these skills describe social relations or interactions, and they are considerably more general than those used in our earlier work on single-agent plan understanding. Table 2 gives the heads of the 13 social rules that we provided to UMBRA for reasoning about the fable scenarios.

4.2 Processing Extensions

Although these representational extensions provide UMBRA with the content necessary for social understanding, it also required augmentations to its mechanisms to take advantage of them. These changes revolve around processing information about times and constraints, as well as reasoning over the embedded structures used to encode models of others' mental states.

The most basic extension concerns time. In particular, when UMBRA makes inferences, it provides the inferred beliefs with start times based on the current cycle. It also generates constraints stating that the start times for any embedded elements occur before their associated end times. The system furthermore adds constraints to working memory as default assumptions when they appear as rule conditions and they are not already present in memory. When the latter include unbound variables, it inserts Skolem values to denote unknown times that satisfy the specified temporal relations. The system does the same for other constraints involving equality or inequality.

UMBRA also takes constraints into account when making inferences. In particular, the system eliminates from consideration any rule instance that would produce a default assumption inconsistent with any temporal or other constraint already in working memory. For example, if the constraint *before(t1, t2)* is present and a rule instance would add *before(t2, t1)*, then it would skip this candidate and consider other alternatives instead. As a result, the system never produces an explanation with such direct contradictions, although indirect ones can arise, as we discuss in Section 7. This ability also improves efficiency, as the system identifies low-value branches in the tree of possible explanations and prunes them.

The other major processing extension involves giving UMBRA the ability to reason over nested expressions. Most domain rules refer to believed or desired relations in the external world, but the content against which they match may be embedded within models of others' mental states. For instance, the system may have a rule that includes an antecedent *can_eat(Agent, Substance)* but have in working memory *belief(snake, belief(fox, can_eat(crow, grain), t1, t2), 09:50, t4)*. To align non-embedded rules with such elements, the augmented UMBRA strips both rule antecedents and working memory elements of their wrappers to expose their domain-level content. If the system finds a potential match, it selects the appropriate level of nesting to unify antecedent with element, then translates the entire rule to this level by wrapping its antecedents in the notation for models of mental states. Although in principle such embeddings can be arbitrarily deep, our current implementation only adds up to three levels of nested mental states.

UMBRA did not require any changes in its abductive inference mechanisms to utilize the knowledge about social interactions shown in Table 2, despite their more abstract character. During processing, the system unifies the variables in these rules with the concrete structures that arrive as input and that result from domain-level inference, incorporating them into the explanation that it constructs incrementally. Instances of these rules provide connective tissue that make social explanations more coherent, and thus aid the overall abduction process. The fact that no changes to the process were needed does not reduce the importance of abduction to our account of social understanding – it only shows the generality of the existing mechanism.

4.3 Central Features

We can summarize these representational and processing extensions in terms of three capabilities that appear to be central to the task of social understanding:

- Inference over domain-independent social knowledge based in large part on mental structures;
- Mental states with intrinsic temporal qualities, including constraints on the latter; and
- Reasoning over different levels of embedding using the same knowledge elements.

We believe these interrelated abilities are necessary for mental state ascription through 'everyday' reasoning processes such as abduction. A system that lacks the first would not link agents' mental states to their behavior. Reasoning without the second could not explain nuances such as the difference between an agent having a belief *after* or *until* they were informed of something. The third allows for arbitrary depth of application without requiring a new rule for each model.

Although these features are not innovative in themselves, we believe our combination and application of them is novel.[4] We do not claim that they are the only important capabilities required for social understanding, nor that ours is the only reasonable approach to all tasks of this nature. For example, Polyscheme (Cassimatis, 2006; Bello, 2012), in its search through possible worlds during default reasoning for mental state ascription, utilize the first two ideas, while the need for the third is obviated by use of multiple 'worlds' and operations that support reasoning across them.

5. Empirical Evaluation

We have described some extensions to UMBRA that were motivated by the task of social plan understanding, but whether they work as intended remains an empirically verifiable question. In this section, we present a number of claims about the system, the dependent variables we have used to measure its behavior, and some experiments designed to test those claims.

5.1 Claims and Methodology

We are interested in UMBRA's ability to weave an explanation of observed or reported social interactions. This should include not only simple interchanges, but ones that require the participating agents to reason about others' mental states, including situations that involve ignorance (e.g., mistaken beliefs and faulty reasoning) and behaviors that capitalize upon that (e.g., opportunism and deception). We are thus chiefly concerned with UMBRA's capacity for reconstructing an accurate and complete account of events based on the partial information provided to it.

We can transform these ideas into three empirical claims or hypotheses about the extended system's ability to understand social behavior:

- The extended UMBRA can generate appropriate explanations and inferences, for the fables described earlier, from partial information;
- The system's ability to apply its knowledge at different levels of embedding is critical to this functionality; and
- High-level knowledge about social interactions is also essential to UMBRA's ability to generate reasonable social explanations.

The first hypothesis relates to the system's basic capability for social understanding, whereas the others concern the benefits of two extensions we introduced for this purpose. To test these claims, we designed and ran a number of experiments. In each case, we ran UMBRA on each of the eight scenarios, presenting it incrementally with a succession of observed events.[5]

4. However, note that AbRA (Bridewell & Langley, 2011), an earlier abductive system, incorporates very similar ideas.
5. Initial studies suggested that running the system in "batch mode" by providing the inputs up front only improved its performance a little, which is consistent with our earlier results on single-agent plan understanding. Thus, we limited our formal experiments to incremental processing, which seems closer to the ways humans operate in real settings.

Table 3. A selection of inputs and desired outputs for a small part of the scenario 'The Fox And The Crow'. The outputs shown are those used in rules for offering a trade in bad faith and for trading food (as applied by the crow). Constraint elements are not shown. Skolem values are given in the form *sx*.

Elements given:

belief(observer, belief(crow, exists(grain1, 08:00, 24:00), 08:01, _), 08:01, _)
belief(observer, belief(crow, has(crow, grapes1, 08:00, *s1*), 08:01, _), 08:01, _)
belief(observer, belief(crow, exists(grapes1, 08:00, 24:00), 08:01, _), 08:01, _)
belief(observer, belief(crow, not(dead(crow, 08:00, 24:00)), 08:01, _), 08:01, _)
belief(observer, belief(crow, okay(grain1, _, _), 09:00, _), 09:00, _)
belief(observer, belief(crow, has(fox, grain1, _, *s1*), _, _), 09:03, _)
belief(observer, belief(crow, not(okay(grapes1, 08:00, 24:00)), 08:00, _), 09:03, _)
belief(observer, belief(crow, suggest_trade(crow, grapes1, fox, grain1, 09:01, 09:03), 09:04, _), 09:04, _)
belief(observer, belief(crow, has(crow, grain1, 09:05, _), 09:05, _), 09:05, _)
belief(observer, belief(crow, actually_trade(crow, grapes1, fox, grain1, 09:03, 09:05), 09:05, _), 09:05, _)
belief(observer, belief(crow, has(fox, grapes1, 09:05, *s2*), 09:05, _), 09:05, _)
belief(observer, belief(crow, not(has(fox, grain1, 09:05, _)), 09:05, _), 09:05, _)
belief(observer, belief(crow, not(has(crow, grapes1, 09:05, _)), 09:05, _), 09:05, _)

Desired inferences:

belief(observer, belief(crow, agent(fox), 08:01, _), 08:01, _)
belief(observer, belief(crow, agent(crow), 08:01, _), 08:01, _)
belief(observer, belief(crow, has_attribute(crow, is_scoundrel), 08:01, _), 08:01, _)
belief(observer, belief(crow, can_eat(crow, grain1), 09:00, _), 09:00, _)
belief(observer, belief(crow, belief(fox, okay(grapes1, _, *s2*), 09:00, _), 09:00, _), 09:00, _)
belief(observer, belief(crow, belief(fox, can_eat(fox, grapes1), 09:00, _), 09:00, _), 09:00, _)
belief(observer, belief(crow, goal(fox, trade(crow, grapes1, fox, grain1, _, _), 09:03, _), 09:04, _), 09:04, _)
belief(observer, belief(crow, suggest_trade_bad_faith(crow, grapes1, fox, grain1, 09:01, 09:03), 09:04, _), 09:04, _)
belief(observer, belief(crow, trade(crow, grapes1, fox, grain1, 09:01, 09:05), 09:05, _), 09:05, _)

The inputs on each run comprise those elements of a fable, stated in logical form, that are available to the observing agent, along with background facts about the domain. Elided elements are those unobserved actions and unknown concepts that are not explicitly available in the story. Thus, the input includes no constraints or goals, and only some of the beliefs at appropriate levels of embedding. The number of elements elided from the input trace varies across scenarios; in total, we provided about 37 percent of the desired explanations to the system, forcing it to infer the rest. Table 3 presents some of the inputs provided for The Fox and the Crow fable, along with a sample of the inferences that we wanted UMBRA to make in response.

5.2 Dependent Measures

We cannot evaluate UMBRA's abilities without some measure of its behavior. The most natural dependent variables are *precision* and *recall*, which are widely used in research on natural language, especially information retrieval and question answering. Briefly, the system should neither omit plausible conclusions nor draw inappropriate ones. For each fable, we enumerated a set of ground literals that we believed should be inferred given the observations. For each run of UMBRA on that fable, we counted the number of inferred elements that appeared in this target set (*true positives* or

TP), the number of inferences that were not in the target set (*false positives* or *FP*), and the number in the target set that the system failed to produce (*false negatives* or *FN*). We then combined these to calculate precision ($TP/(TP + FP)$) and recall ($TP/(TP + FN)$).

Our metrics are similar to, but differ in detail from, metrics used in other recent efforts on abductive inference. As in Bridewell and Langley's (2011) work on plan understanding, we measure precision and recall over all elements in the explanation. This contrasts with the scheme that Raghavan and Mooney (2010) report, which focuses only on the top-most literal in a hierarchical plan, and the one that Kate and Mooney (2009) use, which deals only with inferred nonterminal elements. We maintain that measuring precision and recall over all literals not provided as input is more appropriate for tasks that involve plan understanding. These should include literals that describe states, not only those that refer to activities, as typically done in the plan recognition literature.

Our measures are more stringent that those of Raghavan and Mooney (2010), in that we regard any inferred literal with *some* incorrect content as entirely wrong, rather receiving partial credit based on its predicate and some arguments. We also diverge from Bridewell and Langley (2011) in that we automatically count literals with Skolems as incorrect unless the observations provide no way to make them constants, in which case they are acceptable provided their use is consistent with the target literals. We additionally count unobserved temporal constraints in our measures, since these constitute key parts of explanations.[6]

5.3 Basic Explanatory Ability

Our first claim was that UMBRA can generate appropriate explanations from partial information about scenarios that involve social interaction, including ones that involve misunderstanding and deception. Our main approach to testing this did not involve a controlled experiment, but instead was an instance of *demonstrating new functionality*, which Langley (2012) has argued is an important form of evaluation in cognitive systems research. To our knowledge, no previous work on plan understanding has demonstrated this capability, making progress here an important achievement.

Figure 1 (a) shows the precision and recall scores for the extended UMBRA on each of our eight fable-like vignettes. The number of target inferences ranged from 34 to 141 per scenario, with a mean of 73.3, and involved between zero and four levels of embedding. The graph reveals that the system generally has high precision, meaning it generates few undesired inferences, as well as high recall, meaning that it produces most of the desired inferences. In general, UMBRA's reasoning mechanism generates few false positives and few false negatives. Moreover, additional runs suggested that altering the maximum number of assumptions allowed per rule, the main system parameter, had little effect on these scores – slightly increasing false positives but also increasing true positives, as reflected in the higher recall and lower precision scores in Figure 1 (b).[7]

6. There were a small number of cases in which UMBRA made inferences that were orthogonal to the target explanation but still seemed valid. For instance, the observer believed the eel was wet in a scenario that involved the eel living in a pond. These 'non-canonical' inferences were excluded rather than being counted as true or false positives. The fact that our ground truth measure was defined by the system developers may draw claims of bias; however, there is a dearth of other options, and this approach has been prevalent in earlier work of this type. A possible alternative might be to have results evaluated by a panel of judges.
7. Varying this parameter did affect UMBRA's inference time, with higher settings (the maximum being six) slowing the system's behavior substantially, suggesting an important area for future research.

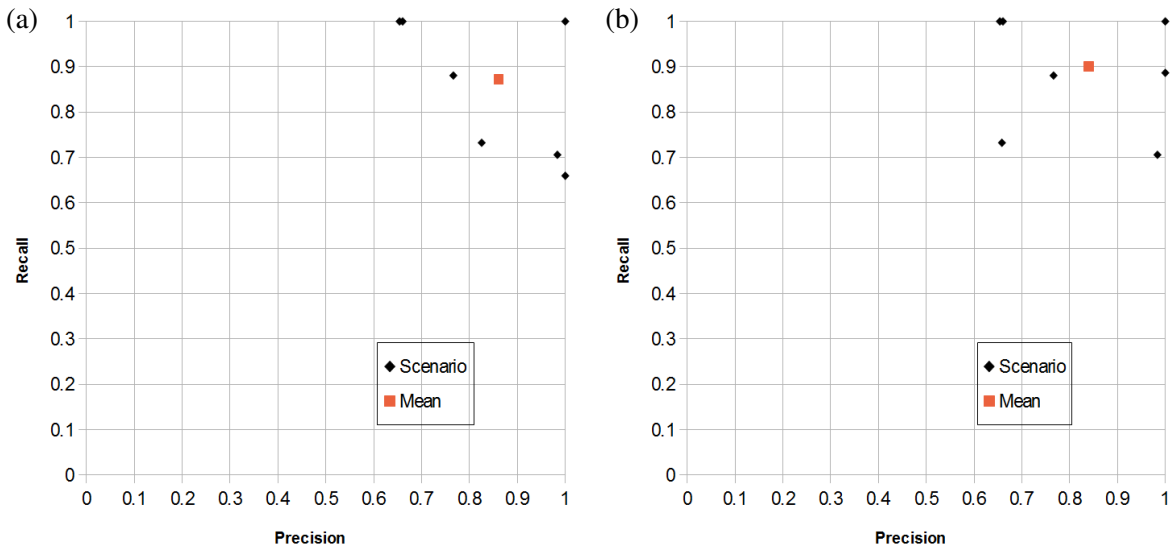

Figure 1. Precision and recall scores on each of the eight fable scenarios discussed in the text for (a) runs of the extended UMBRA with partial information and (b) the same runs with the number of assumptions allowed per rule increased from four to six.

UMBRA had difficulty with a few of the scenarios. For instance, one of the rightmost points in Figure 1 (a) corresponds to an instance of perfect precision but imperfect recall, in that the system omitted some target inferences. In this vignette, The Fox and the Crow, the system infers the observer's beliefs, the observer's beliefs about the crow's beliefs, the observer's beliefs about the fox's beliefs, and the observer's beliefs about the crow's beliefs about the fox's beliefs. Several elements of the exchange were missed at this last, deepest level – for example, the belief that the crow believes that the fox believes that the crow suggested a trade in good faith. This is likely because there is typically less information available at deeper embeddings.

Conversely, the leftmost points correspond to instances of perfect recall but imperfect precision. For example, in The Hungry Crow, the system initially explained the crow's journey to the barn where it knew the jar of grain was kept in terms of the crow 'visiting' the jar. This part of the explanation was competing with the true "acquire edible food" explanation, which at this early stage in the sequence of observations was more difficult to infer. Specifically, beliefs about the crow opening and closing the jar, the grain being in the jar, and the grain being edible to the crow, were more costly to assume than the agency of the jar and the crow's judgement that the jar was not hostile – all that was required for the crow to be visiting. At this early stage in the run, where plenty of computational resources were available but inputs to incorporate into the explanation were still sparse, the system was able to conclude that the jar ate the crow, and that the crow therefore made a poor judgement call. Upon incrementally processing later inputs about the crow's actions in the barn, UMBRA reconstructed the entire target explanation correctly, but these extra incorrect details remained in working memory.

In the case of the two scenarios that did less well on both precision and recall – The Lurking Eel and The Trapped Crow – both types of error appeared. Some of the social interactions went

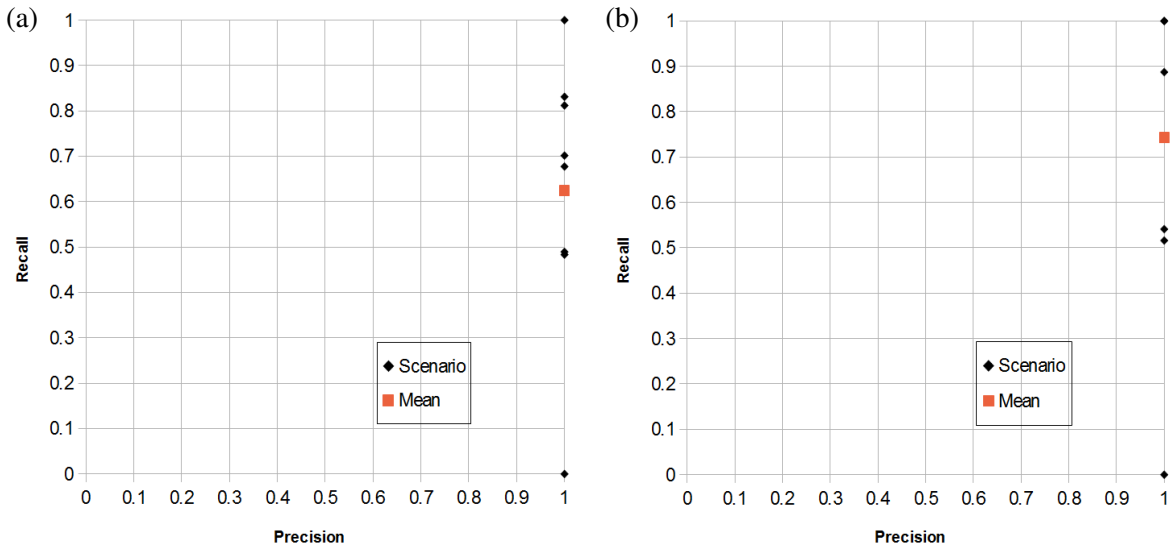

Figure 2. Precision and recall scores on each of the eight fable scenarios discussed in the text for UMBRA runs, after removing (a) the ability for embedded processing and (b) knowledge about social interactions.

unexplained ("the crow believed that the fox believed the crow asphyxiated") and, at the same time UMBRA jumped to conclusions ("the fox successfully dragged the eel out of the pond"), or was excessively imaginative when it could later have explained things better using new supporting input ("rather than escaping, the crow left the fox's den in order to lay in ambush for the fox").

Recall that we are centrally concerned with UMBRA's ability to understand social interactions from only partial information, so these initial runs included less than half of the leaf literals in the target explanations. We measured precision and recall only over these elided elements and target constraints, which we also omitted from the input.[8] However, in order to establish reasonable behavior, we also ran the system again with all leaf literals (other than constraints) for each target explanation given as inputs, with a very low threshold on assumptions per rule application. This let UMBRA operate in an effectively deductive fashion, since each rule application involved inferring only the head, as the body elements were already present. This time, the system achieved perfect precision and recall on each of the eight fables. The system's perfect behavior under these conditions was fully expected, but served to confirm that UMBRA's abduction mechanism reaches the same conclusions as deductive reasoning when it must infer missing elements as default assumptions.

5.4 Utilizing Embedded Structures and Social Knowledge

These results are encouraging, but we should also show that the source of UMBRA's ability to handle social scenarios come from the recent extensions. To this end, we ran the system under two additional experimental conditions in which we removed them. In each case, we provided the same

8. Note that, in practice, UMBRA made many correct inferences involving assumptions about elements which would in fact later appear in the inputs. These additional successes are not recorded in the measurements given in this paper, so the figures actually underestimate UMBRA's reasoning capabilities.

input information as in the second, 'deductive' condition above, in that we provided UMBRA with literals for each of the terminal nodes in the target explanation for a given fable. The purpose was to eliminate interactions between the abductive aspects of UMBRA's operation and the lesioned mechanisms. While we could have studied the lesioned system's behavior in abductive mode, this would not have addressed our questions as well.

We first eliminated UMBRA's ability to reason over embedded structures in working memory, effectively removing its capacity to imagine that another agent might apply the same domain rules it reasons over, and thus to draw analogous inferences. By removing the nesting mechanism, we restricted the system to using rules only at the level of the primary agent's direct inferences about reality. Figure 2 (a) presents the results on the eight fable scenarios. As expected, the perfect precision is unaffected, since eliminating nested inference only reduces the number of conclusions drawn. In contrast, recall drops to a mean of 62.4%.

Our second manipulation involved removing UMBRA's access to the knowledge about social interactions presented in Table 2, which we expected to be critical to properly understanding the fables. Each of these 13 rules incorporated multiple interacting agents, referred to mental states of those agents, and included at least one element that referred to domain-independent content (such as the false information in a deceptive act). As noted earlier, these rules included knowledge about the effects of communicative acts, jealousy, deceptive behavior (e.g., playing dead), offering trades in good or bad faith, viewing (rightly or wrongly) others as threats or nonthreats, and perceiving others through images (e.g., reflections in water). Figure 2 (b) shows the results of running UMBRA with this diminished knowledge base. As before, precision is maintained because the lesion strictly reduces the number of inferences generated, but again recall drops, this time to a mean of 74.3%.

The first set of results supports the second claim presented earlier, that much of UMBRA's ability to understand social interactions depends on applying domain knowledge at different levels of mental representation. The second study's findings support our third hypothesis, that social understanding benefits from the application of high-level knowledge about social interactions. These results are not especially surprising, but they underpin the deeper claim that social cognition depends centrally on the ability to reason about others' mental states. One can imagine other representations and mechanisms that support this capacity, such as the "worlds" framework that Bello (2012) proposes, but our results show that UMBRA embodies a viable response to this challenge.

6. Related Research

As we have noted, our approach to social understanding relates to many earlier themes in AI and cognitive psychology – too many to review in detail. However, our research draws on three main ideas to account for social cognition, and in this section we examine prior work on each of these.

One central assumption in our work in that *social cognition relies on representing and reasoning about models of other agents' mental states*. A number of other researchers have described systems that adopt this idea. Fahlman's (2011) Scone encodes models of mental states in terms of 'contexts' or 'worlds' that separate memory into distinct partitions; his system also includes the ability to comprehend deceptive activities by applying knowledge to content in nested contexts. Polyscheme (Cassimatis, 2006; Bello, 2012) adopts a similar approach, organizing content into worlds that let it carry out counterfactual reasoning and mind reading for purposes of behavior explanation. Both

Scone and Polyscheme rely on default reasoning supported by inheritance, which UMBRA achieves through nested application of domain rules. Another example is Bridewell and Isaac (2011), who introduced a computational framework for deception based on the capacity to reason about the goals of other agents, resting on mental state ascription. Again, their account stores mental states in partitions, with content moved between them to deal with conflicting beliefs. Our work shares several tenets: explanations are constructed in an incremental fashion, different forms of deception involve associated patterns of mental states, and plausible assumptions are useful for related inference tasks.

Our second key assumption is that *plan understanding involves a process of incremental abduction that constructs an explanation of observed inputs*. We have borrowed this idea from AbRA (Bridewell & Langley, 2011), which we view as UMBRA's theoretical predecessor. Although both systems construct cohesive explanations using a form of incremental, data-driven abduction, they differ in many specifics, including their approaches to constraints and time. UMBRA unifies Skolems automatically during inference for purposes of explanation, whereas AbRA makes explicit decisions for this purpose. Both use a form of lookahead when evaluating rule instances, but their criteria for selection differ markedly. For example, UMBRA views the rule choice metric as a 'cost' function and links it to the number of rule applications allowed on a given input cycle. Other work on abductive inference for plan understanding (e.g., Kate & Mooney, 2009; Raghavan & Mooney, 2010) are more distantly related, in that they operate in a query-driven and nonincremental fashion.

A final assumption of our work is that *social understanding depends not only on domain knowledge, but on more generic knowledge about social interactions and their effects on mental states*. Wilensky (1978) reported early research along these lines: his PAM system inferred the intentions of interacting agents, but its reasoning was shallow compared to that in UMBRA. More recently, Winston's (2012) approach to story understanding incorporates reflective patterns, describing abstract social relationships, to support inference and question answering. However, his Genesis system does not appear to include knowledge about how agents' actions influence others' mental states.

Our approach to social cognition incorporates ideas from earlier research traditions, but combines them in novel ways to support new capabilities for deep understanding of scenarios in which agents reason about each others goals and beliefs. Our theory has much in common with Castelfranchi's (1998) framework for social intelligence, in assuming that agents interact with others to achieve their goals and that mind reading plays a key role in this process. However, where Castelfranchi's contribution was purely theoretical, we have grounded our ideas in an implemented system.

7. Concluding Remarks

In this paper, we have introduced and addressed the task of social understanding. We gave a theoretical basis for social cognition and formalized it in the context of a number of relevant fields, including activity and plan recognition, behavior explanation, collaborative planning, and story understanding. We suggested fables as good case studies for this endeavor, because of their focus on goal-directed social interactions with agents modeling each others' mental states.

We described UMBRA, an implemented system for incremental abductive plan understanding, and laid out the representational augmentations and processing extensions necessary for representing and explaining the plans of social agents. The improved system makes inferences not only about

the environment and agents in it, but about agents' mental states as they apply the same knowledge it has available. We carried out experiments with our system on eight fable scenarios, demonstrating its ability to generate appropriate explanations from partial information, achieving precision and recall scores of over 85 percent. We discussed how our approach innovatively combines other threads of research to support deep understanding of social situations.

There are several directions in which we might extend UMBRA in future work. One obvious addition would be a broader body of high-level social knowledge, encoding behaviors such as *instruction* (realizing an agent lacks a relevant belief and providing it), *rectification* (recognizing and correcting an agent's false belief), and *persuasion* (goal-directed action to bring about an agent's application of some reasoning rule). This may involve new types of knowledge, and the capacity to deal with differences in knowledge and disposition between agents. The architecture itself could also be extended to model other facets of social behavior, such as morality and emotions (two aspects of the fable setting we have not yet addressed).

We are planning several immediate extensions to UMBRA. One involves improving efficiency by learning rule embeddings after generating them, instead of repeated generation. Changes to mental embeddings will let agents *introspectively* identify and reason about their own mistakes, as well as infer beliefs that they have particular beliefs. We will add goal-generating knowledge and meta-level mechanisms for goal ascription and inference of causal links between agents' goals and activities, bringing us closer to the traditional focus of work in this area. This will also lead to support for question answering and autonomous learning of social skills and conceptual rules.

We intend to replace standard unification with statements of explicit identity constraints, so that an agent may (for example) cease to believe that two things are the same. The extended system will also support knowledge about criteria for concluding uniqueness and identity. These should lead to improved contradiction detection, constraint handling, and consistency checking, which will together support a belief revision system that lets UMBRA use new information to identify incorrect default assumptions and rule applications, then make suitable repairs to the explanation.[9]

We have presented a theoretical framework for social understanding and an initial implementation. We have argued that their key components include inference over domain-independent social knowledge, temporally situated mental states, constraints as explicit mental structures, and application of rules at different levels of agents' embedded models. We made clear claims about UMBRA's capacity for social explanation using partial information, empirically demonstrating that this ability depends on the availability of high-level social knowledge and the indirect embedding of rules. Although our work remains in its early stages, we have provided a promising account of social understanding, reported encouraging results, and identified important avenues to explore in the future.

Acknowledgements

This research was supported in part by Grant N00014-10-1-0487 from the Office of Naval Research. We thank Paul Bello, Will Bridewell, Nick Cassimatis, and Alfredo Gabaldon for discussions that influenced the approach to social understanding we have reported here.

9. In social settings, there can be major points of realization or reversal, where one model (for example, "the lion is sick") is replaced by another ("the lion is pretending to be sick").

References

Aggarwal, J. K., & Ryoo, M. S. (2011). Human activity analysis: A review. *ACM Computing Surveys*, *43*, 16:1–16:43.

Bello, P. (2012). Pretense and cognitive architecture. *Advances in Cognitive Systems*, *2*, 43–58 .

Bond, A. H. (2002). Modeling social relationship: An agent architecture for voluntary mutual control. In K. Dautenhahn, A. H. Bond, L. Cañamero, & B. Edmonds, (Eds), *Socially intelligent agents – Creating relationships with computers and robots*, 29-36. Kluwer: Boston.

Bridewell, W., & Isaac, A. (2011). Recognizing deception: A model of dynamic belief attribution. *Proceedings of the AAAI Fall Symposium on Advances in Cognitive Systems* (pp. 50–57). Arlington, VA: AAAI Press.

Bridewell, W., & Langley, P. (2011). A computational account of everyday abductive inference. *Proceedings of the Thirty-Third Annual Meeting of the Cognitive Science Society* (pp. 2289–2294). Boston: Cognitive Science Society.

Cassimatis, N. (2006). A cognitive substrate for achieving human-level intelligence. *AI Magazine*, *27*, 45–56.

Castelfranchi, C. (1998). Modelling social action for AI agents. *Artificial Intelligence*, *103*, 157–182.

Fahlman, S. (2011). Using Scone's multiple-context mechanism to emulate human-like reasoning. *Proceedings of the AAAI Fall Symposium on Advances in Cognitive Systems*. Arlington, VA: AAAI Press.

Goldman, R. P., Geib, C. W., & Miller, C. A. 1999. A new model of plan recognition. *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence* (pp. 245–254). San Francisco: Morgan Kaufmann.

Langley, P., Laird, J. E., & Rogers, S. (2009). Cognitive architectures: Research issues and challenges. *Cognitive Systems Research*, *10*, 141–160.

Malle, B. F. (1999). How people explain behavior: A new theoretical framework. *Personality and Social Psychology Review*, *3*, 23–48.

Meadows, B., Langley, P., & Emery, M. (2013). Seeing beyond shadows: Incremental abductive explanation for plan understanding. *Proceedings of the AAAI Workshop on Plan, Activity, and Intent Recognition*. Bellevue, WA: AAAI Press.

Meehan, J. R. (1977). TALE-SPIN: An interactive program that writes stories. *Proceedings of the Fifth International Joint Conference on Artificial Intelligence* (pp. 91–98). Cambridge, MA.

Mueller, E. T. (2002). Story understanding. *Encyclopedia of cognitive science*. Wiley Online.

Nau, D. S., Cao, Y., Lotem, A., & Muñoz-Avila, A. (2001). The SHOP planning system. *AI Magazine*, *22*, 91-94.

Raghavan, S., & Mooney, R. (2010). Bayesian abductive logic programs. *Proceedings of the AAAI-10 Workshop on Statistical Relational AI* (pp. 82–87). Atlanta: AAAI Press.

Rao, A. S., Georgeff, M. P., & Sonenberg, E. A. (1992). Social plans. *Proceedings of the Third European Workshop on Modelling Autonomous Agents in a Multi-Agent World* (pp. 57–76).

Wilensky, R. (1978). Why John married Mary: Understanding stories involving recurring goals. *Cognitive Science*, *2*, 235–266.

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