1 Introduction

Instruction has typically inspired rote, exemplar-based, and explanation-based learning techniques; but learning from instruction is not merely mimicry, or even generalized mimicry. A teacher can supply not only examples, but can point out relevant features, indicate task structure, demonstrate alternative approaches to subtasks (including approaches that will fail), and generally guide the student through the process of constructing a generalized procedure for the task. The machine learning approaches mentioned above fail to take advantage of, and are actually foiled by, such communicative acts. The aim of my research is to investigate a computational approach to understanding instruction of this nature, and thus more fully exploit the guidance that a teacher can supply.

The underlying assumption in this work is that the instructor generates explanations felicitous to learning [Van83]. The learning system will exploit the bias provided by these felicity conditions to guide the search for an interpretation of the teacher's actions as a coherent explanation, and then use this explanation to construct a generalized domain plan. As in [Lit85] two levels of plan recognition are used to explain the teacher's behavior: domain and discourse. Domain plan recognition follows the standard model of decomposition hierarchies [Kau90], and is used to identify segments of instruction which correspond to known domain operations. In this research the primitive level of this hierarchy consists of move, open, and close operations for a robot arm. Higher levels of the domain hierarchy consist of compositions of operators at the level below. Consider the task of constructing an arch: two columns topped by a lintel. The columns must be of equal height and must be separated by an amount equal to the length of the lintel. An instructor may communicate the task of constructing an arch by picking up the lintel that will span the columns and placing it directly beside the location where the columns will be constructed. This action is not part of the task itself, but is crucial in the communication process — it identifies that the lintel should be in the focus of attention. The instructor may then place the two foundation blocks and lay the lintel on top: an arch. To communicate that the columns of an arch may be arbitrarily tall the instructor may then remove the lintel, place one more block on either column, and replace the lintel; repeating a few times will express multiple examples of arches, as well as the process by which the concept may be extended. Negative examples may be communicated in a similar manner: the instructor may build one column taller than the other, fail to place the lintel, and then correct the problem. Supposing that the learner was familiar with the tasks of moving blocks and lintels, and constructing towers it would be able to recognize a portion of the instruction as domain operations, but to the learner these domain operations do not compose to any known operation, and so in terms of domain plan recognition they are incoherent. Discourse plan recognition is used to understand how such segments form a coherent explanation. No generalized theory of plan recognition for discourse plans currently exists. Litman's stack based scheme for recognizing discourse plans was developed to explain interruptions and clarifications, and does not seem capable of explaining more general discourse phenomena such as those discussed in the preceding example. The recognition of discourse plans in the instructional setting comprises the core of this research.

The following section contains summaries of a variety of papers which appear in the proceedings, as well as one of the invited talks. A discussion of how these ideas relate to the problem of understanding instruction is integrated into each summary, as well as forming a conclusion for the report.

2 AAAI-91

To the best of my knowledge the use of discourse understanding for machine learning is novel. That is to say do not be surprised if the following summaries of *related work* from AAAI-91 do not bear an obvious relation. The following sections of the proceedings contain papers which I consider relevant: generation and interaction, syntax and semantics, formalisms for coordination, temporal constraints, tractable inference, and an invited talk on discourse understanding. A further invited talk by Diettriech gave an excellent summary of the machine learning field, but was otherwise irrelevant except for a passing mention of the need to explore, among others, the area of learning from instruction; and so it will not be summarized or discussed. The summaries of the various papers (and talk) below include a brief discussion of their relevance to my research.

2.1 The Computational Understanding of Discourse

— Candice Sidner

The problem of understanding instruction is a subproblem of understanding dialogue in general. Sidner's unpublished invited talk provides an overview of the field of discourse processing, distinguishes it from natural language processing, and indicates some important future directions for research. The view of discourse presented here is more general than need be for the current form of my research, and as these differences are encountered I will point them out.

Sidner begins by defining discourse in terms of its usage: it is used by intelligent agents to collaborate, negotiate, and communicate intentions and beliefs. To further distinguish discourse from natural language she notes that discourse places utterances in context, discourse may contain multiple utterances, and discourse considers multiple participants. I will inject at this point that although there are multiple participants in a discourse there need not be utterances from more than one. The other participants are considered in that their mental states are affected by the speaker, and their mental state must be considered when analyzing the reason for the utterances.

The history of discourse processing research begins with the philisophical views of Austin[Aus62], Grice [Gri68], and Searle [Sea69] in which language is seen as a means of manipulating other minds. This is seen as the precursor of the modern view of discourse. In that view the speaker would implicate, and the listener's task was to infer the speaker's meaning. Austin began this line of research by attempting to classify the kinds of things that could be done with utterances, how the various categories could be recognized, and to what extent context was involved. He drew the analogy between speech and action which would later inspire Searle's formulation of "speech acts". The computational approach to discourse understanding arose from this action analogy — a speaker was considered to be executing a plan, and the listener's task was to recognize that plan. Perrault, Cohen and Allen [CP79] first implemented this model. Numerous refinements have been made to this model including Litman's meta-schemas for recognizing interruptions and clarifications, and Grosz's global structure of discourse in which plans were organized around the agents task. I will note at this point Cawsey's explanation generation system uses both of these ideas though the application is inverted.

Sidner's model of discourse is composed of three distinct and interacting components: linguistic, intentional, and attentional structures. The linguistic component is very close to a natural language parser, but also makes special note of surface linguistic clues as in Hobb's (1977). The attentional component is along the lines of Grosz's [Gro77] focusing semantic net in which parts of the plan were distinguished as subtasks by the domain knowledge that they focused upon. The third component, the intentional structure captures prepositional attitudes including goals and subgoals, as well as the more complex notion of intentions toward other agents. Sidner makes no mention of beliefs with respect to this model, but her mention of Pollack would seem to indicate that her notion of the intentional component was similar to

Pollack's which captured both belief and intention. Certainly beliefs must be included in the model. My model has them as a separate fourth component.

The introduction of beliefs into the model is problematic; how do we handle discourse between two agents with different beliefs? Earlier models of discourse processing in which agents shared beliefs, and only had to infer intentions, had no such problems. In those systems, if an agent performed some operation, then it would imply certain things about the agent's intentions. But if these agents have different beliefs (e.g. different task knowledge) then the hearer may ascribe the wrong intentions to the speaker. For Sidner this problem is solved by a process of collaboration which is itself part of the discourse. Agents collaborate on both activities and establishing shared intentions and beliefs. This collaboration is evidenced by clarification requests, turn taking procedures, reference formulations, etc. In terms of activities, collaboration is used to establish common goals, what each agent should do, and how. In this model of discourse and activity, planning and acting are highly interleaved. Planning is incremental, both planning and acting are distributive, and each agent may have only partial knowledge.

In the instructional setting the shared belief model makes no sense at all. The assumption is that the agents have different beliefs, hence the need for instruction. The purpose of the entire discourse is to build a shared belief structure about some task. Collaboration in my model is currently rather one–sided: the instructor is the only producer of utterances, and as such has the responsibility to communicate his/her intentions and beliefs to the student. In the instructional domain this assignment of responsibility seems intuitively correct, but certainly more complex interactive collaboration (e.g. interrupting clarifications) should be accommodated.

2.2 Generating Interactive Explanations

— Alison Cawsey [Caw91]

In my research the instructor can loosely be thought of as generating explanations; the instructor must explain how a goal can be accomplished in a generalized setting. Cawsey's paper describes a computational approach to generating such explanations and descriptions. Her approach is based on text planning, discourse analysis, and user modeling.

Discourse analysis has guided Cawsey's approach by showing that formal types of dialogue (e.g. classroom, courtroom, etc.) have a regular hierarchical structure [SC75]. The details of the structure depend on the type of dialogue rather than the domain content. Cawsey's model of plan generation is based on incremental hierarchical planning. The content of the interaction is planned for separately from the dialogue itself. Dialogue is planned to achieve maximal coherence with the discourse context which includes, among other things, information about what concepts are currently in focus. The content of the dialogue is planned based primarily on a user model. Portions of a standard explanation which the user is known to already understand may be skipped, and user knowledge may be exploited to compare or contrast with another concept. The system may query the user when the user model cannot indicate whether or not the user already has knowledge about the subject matter. User interruptions may modify the discourse context and/or the user model, so planning is incremental allowing the course of the dialogue to be appropriately modified. My research currently ignores the interactive aspect of student/teacher dialogues, but future extensions should certainly reintegrate this aspect.

Two types of relation exist between sections of the explanation at the content level. First, coherence relations attempt to keep related topics grouped. Second, prerequisite and subskill relations attempt to define an order in which related material should be presented. Cawsey defines content operators designed to capture these relations. For example, in her domain she defines an operator to explain how a device works, but it has as preconditions that the student must know the parent class of the device, the function of the device, and the structure of the device. In my system a plausible prerequisite for teaching subtask **A** might be that any subtasks of **A** be taught first; and a coherence constraint might be that once these subtasks of **A** have been taught that **A** itself be taught rather than switching to teaching subtask **B**.

Dialogue planning is defined separately from, but in a similar manner to content planning. The levels in the dialogue hierarchy are based on the levels of description found in [SC75]: transactions, exchanges, moves, and acts. A transaction is composed of a particular sequence of exchanges, exchanges of moves, and moves of linguistic acts. Dialogue operators take content goals as arguments, and prescribe how to present the content. Cawsey's dialogue operators capture discourse phenomena such as that an opening boundary consists of an opening marker followed by a meta comment about the future discourse. In my setting both of these must be expressed with actions, so an opening marker may be a do–undo operation or some other alarming pattern, and a meta comment about the future dialogue may take the form of a simple example of the task to be performed.

This research provides a good deal of insight into the schemas and heuristics that guide discourse in the instructional setting. This should be helpful in constructing a model of plan recognition. The dialogue schemas may in fact be transportable, but the content schemas will need complete revision, and the difficulty of determining if a content goal has been achieved should not be underestimated. The obvious gap is that the instructor and student will not have shared knowledge. Still the coherence heuristics may provide the learner with a strong bias in the search for an interpretation of the instructor's explanation.

2.3 Observations on Cognitive Judgements

— David McAllester [McA91]

In addition to the coherence and prerequisite relations that guide Cawsey's explanation generation system a third class of relations need be considered to influence presentation: obviousness. Two ways of presenting material may be equally coherent and the prerequisites may be fulfilled for both, yet one may be superior to the other in that it may make the task of inference simpler. It seems intuitive that people would adhere to such a bias during dialogue — especially instructors and students. As well as the computational advantage provided by this bias is the cognitive modeling appeal.

McAllester's paper investigates the possibility of constructing analytic theories of *cognitive judgements* — the classification of statements as obvious or not. In particular cognitive judgements about statements whose truth seems to require reasoning involving mathematical induction. Intuitively a statement is obvious if its truth value is judged without an intervening reasoning process. Note however that the correctness of the judgement of the truth value does not affect the obviousness of the statement. An analogy can be drawn between analytic theories of cognitive judgements and analytic theories of grammaticality: a finite grammar can generate an infinite set of grammatical sentences from a finite dictionary, and a finite set of inference rules can generate an infinite set of formulas from a finite set of premises. In general, in linguistics parsers can determine if a sentence was generated by a given grammar, but for inference the process tends to be intractable. However, there is a class of sets of inference rules, *local rule sets*, for which whether or not a statement is generated by the rules can be determined in polynomial time in the length of the statement.

The paper presents two rule sets that include an inference rule for mathematical induction. Both turn out not to be local rule sets, but they do explain a large number of examples tractably; and examples that they fail for are difficult to devise. Although the author was unable to generate a local rule set for mathematical induction, it is claimed likely that a large local rule set could be constructed for the task. McAllester claims that such a rule set could have important practical value as well as providing a formal framework for the construction of linguistic theories of cognitive judgements. From my perspective, if such a rule set could be found then it could be used to guide generation and interpretation of dialogues. In particular, if a local rule set could be found for cognitive judgements about an instructors actions, then it could be used to disambiguate between multiple interpretations.

2.4 A Probablilistic Model of Plan Recognition

— Eugene Charniak and Robert Goldman [CG91]

As suggested above, the instructor can be thought of as generating explanations. Cawsey outlined a plan based approach to this problem. Understanding instruction can then be seen as the problem of plan recognition. A plan recognition system either retrieves or constructs possible explanatory plans based on observations of an agents actions. Kautz's well known formalization of this process is based on minimizing the number of top-level plans. As the authors point out, this approach cannot choose a single plan from the set of top-level plans, no matter how likely one is, and how unlikely the others are. This notion of likeliness is of course very similar to that of cognitive judgements. A second flaw is with the notion of a top-level plan. The problem is two-fold: what we may treat as a top-level plan at one point, we may treat as a subgoal at another; and in a learning system, as in my work, the hierarchy will change over time — possibly during the recognition process. A more serious problem is the obvious fact that the restriction to the minimal number of top-level plans can eliminate the correct interpretation.

The approach advocated by the authors is to reason under uncertainty. That is, to accept all plausible hypotheses, then evaluate the degree to which each is supported by the evidence in order to select the most likely one. This paper outlines a method for choosing the most likely among a set of competing hypotheses. In the instructional setting a student will be presented with information intended to be easily interpreted. Thus, here more than most places selecting the most likely plan is justified. The solution presented here uses a Bayesian network to evaluate the conditional probability of the competing hypotheses based on the observations.

The model of plan recognition consists of a knowledge base of facts about the world including about actions and hierarchical relations between actions, so plan recognition is the standard process of finding the plans for which the observed actions are subactions. The knowledge base also contains facts about the likelihood of events. An inference engine is fed observations, and uses facts from the knowledge base together with a set of forward chaining rules to produce a Bayesian network representing all plausible hypotheses (where plausible is defined by a limit on the probability). At any point during plan recognition the system can evaluate each of the competing hypotheses to choose the most likely one.

While this model of plan recognition does overcome the limitations imposed by the minimality constraint of Kautz, it does not seem to model cognition. People seem to maintain a hypothesis that was deemed likely early during processing, even though later observations may make another more likely. This is probably also observable by labelling the order that events occur in, and then presenting this labelled set in a different order. I would assume that different hypotheses would prevail for different orderings of presentation. A more cognitively plausible model of this process, at least as my introspection dictates, is that the first observation suggests only one plan — the one which is most likely to occur. As each event is observed it is fitted into this plan until the combined probability of the events drops below some limit. At that point a more reasonable alternative is sought based on the observations thus far. Then, as before, future observations are fitted against this new plan until the combined probability again drops below the limit. In fact, even this model seems too lenient. In the instructional setting it is vital that a teacher's plan not be ambiguous. VanLehn [Van90] argues that students who do not recognize the teacher's plan build procedures with bugs, and that these bugs are repaired as opposed to choosing a different interpretation of the teacher's actions.

2.5 Temporal Reasoning During Plan Recognition

— Fei Song and Robin Cohen [SC91]

Plan recognition generally propogates all possible interpretations of actions upward through a schema hierarchy. The hierarchical representation of plans expresses an include relationship between a task and its subtasks, but ignores the temporal relations. As the authors point out, temporal constraints can often be used to eliminate candidate plans during plan recognition. As an example they provide two Chinese cooking tasks that are identical except that the order of two subtasks is switched. Ignoring the temporal information which is readily available from the observations means that both plans would be candidates. In the context of my research temporal reasoning has additional value beyond plan recognition. Plan generalization and reapplication may be able to make use of this information. If it is known that two subtasks can be executed in arbitrary order then a more efficient plan may result. For example, if two arches are to be built, and all of the blocks for the towers are piled ontop of the lintels which are in turn stacked, then if the order of constructing the arches allows interleaving no effort need be wasted; but, if the plan specifies that one arch must be built before the other, then all of the blocks for the second tower might be placed on the table rather than building the second arch's towers with them.

The paper presents an extension to Allen's plan recognition model [All83]. That model constrained the recognition process by checking for inconsistencies between the temporal constraints on plans and those that could be extracted from the input. The approach was a constraint propagation algorithm using intersection and composition over sets of constraints. However, the algorithm for constraint checking did not take advantage of the temporal constraints between an action and its subactions. Allen's algorithm treats all intervals of actions as independent of each other, but for decompositions the interval of the abstract action depends on the intervals of its subactions. The extension presented here by Song and Cohen is based on this idea — the temporal constraints between the subactions in a decomposition are used to compute the boundary intervals on the abstract action. A decomposition is said to be *closed* if the interval of the abstract action is temporally bounded by the intervals of the subaction. Specifically then, the authors present the integration of a closing procedure with Allen's constraint propagation algorithm.

The basic idea of the extended algorithm is to first view the plan as a temporal network and use Allen's algorithm to partially refine the constraints. Then the plan is viewed as a hierarchical structure and traversed in a depth first manner closing all decompositions. This traversal may alter some of the constraints between abstract actions, so the process is repeated until no constraints are changed.

The closure algorithm is actually very simple: a k-ary decomposition is converted into a hierarchy of binary decompositions which is easily closed in bottom-up fashion with the constraints being propagated upward. The closure of binary decompositions is accomplished by dividing the 13 basic temporal relations into five classes, and so any constraint set into five subsets. The five classes define easily solved cases of closure: for abstract action **A** with subactions **a1** and **a2** (1) **A** is bounded by **a1** and **a2** respectively; (2) **A** is bounded by **a2** and **a1** respectively; (3) **A** is bounded by **a1**; (4) **A** is bounded by **a2**; or (5) **A** is bounded by both **a1** and **a2**. By merging the solutions to each of the five closures, and conjoining the clauses, the closure of the decomposition is achieved. This closure may still allow inconsistencies, but usually adds constraints.

Extracting temporal cues from the input is critical to this scheme. In the linguistic domain about which the authors are concerned tense, aspect, temporal adverbials, connectives, and cue-phrases may sometimes be useful for temporal reasoning. In the non–verbal instructional setting in which I work an instructor may indicate temporal relations by, for example, interleaving the achievment of two subgoals to indicate that they can be, or that they need to be, done together; or the instructor may indicate temporal dependence by showing that a certain ordering fails to achieve the goal.

2.6 A Cognitively Plausible Approach to Understanding Complex Syntax

— Claire Cardie and Wendy Lehnert [CL91]

One of the fundamental problems faced in discourse processing is how to update the discourse context. In the paragraph above methods of communicating temporal information were discussed, and it is troublesome to consider how, for instance, the interleaving of subtasks to communicate potential temporal parallelism should affect the discourse context. Grosz's discourse processing system treated the context as a global structure. As a result no form of context switching was accommodated. Thus interruptions and interleaved subtasks would appear incoherent. Litman's system accommodated partial context switches through a stacking mechanism, but only for interrupting subdialogues, and only the linguistic component of the context was switched. Cawsey took Litman's framework and added Grosz's focusing mechanism, but the mechanism remains global and so the problem remains. This paper presents a mechanism for handling complex syntactic structures within a semantic parser. The mechanism relies on a method of context switching when nested structures are detected. The full sense of the word "context" is used here — each structure is parsed in a separate environment. When control is returned to it up to the interrupted process it has the responsibility for updating its own context based on the childs output. Although this idea, in its current form, is still only capable of handling interruptions and not interleaving, it does suggest possible extensions.

The implementation that the authors describe, called CIRCUS, works without a global syntactic grammar, without syntactic parse tree representations, without massive syntactic ambiguity, and without sacrificing the benefits of semantically–oriented parsing. A number of psycholinguistic studies are referenced to show that CIRCUS achieves the desired balance between syntactic and semantic concerns during processing.

CIRCUS is a conceptual analyzer that produces a semantic case frame representation for an input sentence using stack oriented control for syntactic processing and a marker passing mechanism for predictive preference semantics. It uses individual words to selectively trigger one of a small number of lexically–indexed control kernels (LICKs) which will handle the processing of the current embedded clause. It is based on the McEli parser (Schank and Riesbeck, 1981) and uses lexically–indexed local syntactic knowledge to segment incoming text into noun phrases, prepositional phrases, and verb phrases. These constituents are stored in global buffers that track the subject, verb, direct object, indirect object, and prepositional phrases of a sentence. The buffer contents are restricted to simple syntactic structures with a strongly local sense of the sentence. As soon as McEli recognizes a syntactic constituent, that constituent is made available to the predictive semantics module (PSM) which is responsible for making case role assignments. In CIRCUS, this consists of top-down slot filling for any active semantic case frames. Whenever a syntactic constituent becomes available in one of the global buffers, PSM examines any active case frame that expects a slot filler in that buffer, and fills it if the constituent satisfies the slot's semantic constraints. Both hard constraints and preferences are considered. When sentences become more complicated (e.g. wh-constructions), the stack processing is partitioned in a way that recognizes embedded syntactic structures as well as conceptual dependencies. The top-level McEli stack is considered as a single control kernel whose expectations and binding instructions change in response to specific lexical items in the sentence. When a subordinate clause is encountered, the top-level kernel creates a subkernel that takes over to process the interior clause. When processing of the interior clause completes control is returned to the parent. Each control kernel essentially creates a new parsing environment with its own set of bindings for the syntactic buffers, its own copy of the McEli stack, and its own predictive semantics module. The behavior of multiple LICKs is defined by the rules for passing variable bindings between them: (1) when a new LICK is created, all syntactic buffers in the child are intitialized by the parent; and (2) when the child lick terminates, only the parents lick buffer can be modified by the child. LICK's then embody the basic control mechanism of ATN's but enforce a much stricter set of communication rules. In addition, CIRCUS' use of LICKs differs from the pervasive recursion of ATNs in that CIRCUS employs the LICK mechanism only at the clause level and selectively triggers the mechanism via lexically-indexed signals.

Studies in human natural language processing tend to verify the plausibility of the mechanism as a cognitive model. A 1988 study (Swinney et al.) determined that people "reactivate" the meaning of a wh-phrase antecedent at the position of its gap in the embedded clause. This result implies that people have integrated the meaning of the filler into the current semantic representation of the sentence at the point of the missing constituent. CIRCUS demonstrates this reactivation effect when the syntactic constituent currently expected according to the McEli stack is found to contain the antecedent. CIRCUS predicts this effect for antecedents in the subject position, but psycholinguistic experimentation remains to be done to test this prediction. The LICK mechanism also agrees with the results of the Swinney study in that only the correct antecedent is reactivated. CIRCUS is also consistent with evidence for the phenomena known as the filled gap effect (Crain and Fodor 1985, Stowe 1986). Experiments showed an increase in reading time at the point where a postulated gap is found but does not actually occur. The CIRCUS mechanism suggests that this effect is due to semantic reanalysis rather than syntactic as previously suggested.

The LICK formalism may transfer nicely to plan recognition, though the stack mechanism must be revised or discarded. The schema hierarchy might be a reasonable alternative. Rather than allowing a single process to be spawned, the hierarchy could define some legal system of active processes (like a semantic net) which would spawn child and parent processes. The activation order of these processes in the hierarchy could be constrained — a parent process could not complete (recieve a closing discourse phenomena) before its children completed. The activation of a schema outside the tree could signal an interruption, or clarification. The reactivation ideas used in Cardie-Lehnart could be used to control discourse and domain focus.

3 Discussion

There are two aims of the research that I have undertaken. First, from a cognitive science, or strong AI, perspective a model of machine learning which does not exploit instruction as humans do does not comprise a plausible cognitive model. One aim of this research is to develop a cognitively plausible theory of discourse processing for the instructional setting. Second is an applications consideration. Robot programming has begun to be replaced by exemplar-based systems, and mixed systems which use a text skeleton with a teacher to guide the robot through the missing portions, but true learning from instruction capability would be a huge improvement.

From the applications standpoint the ideal goal might be a system which given the goal produces an optimal solution, but an NP–completeness proof for Blocks World planning complexity [GN91] precludes such an approach. Related research depicts the difficulty of learning when few of the attributes present in the environment are relevant to the task being learned [Lit88], and the fact that a teacher can mitigate this computational complexity by indicating which attributes are relevant [Val84]. But a teacher can provide much more information than this. Valiant placed restrictions on the teacher to ensure the legitimacy of his results; restrictions that ensured that the teacher could not encode the procedure that was being learned. But in fact teachers do encode the procedure being learned, and from a practical stance a machine learner should exploit that encoding.

From the cognitive modeling perspective the impetus is clear. No existing theory exists to explain how human learners exploit instruction. Current research in machine learning, almost exclusively, takes the role of the teacher to simply be the provider of examples. More applicable than anything from machine learning is the work in discourse processing — both recognition and generation. Discourse planning in the teacher/student setting can be considered as explanation generation as in Cawsey's system [Caw91]. The schemas, and coherence heuristics she discusses begin to explain the discourse level of planning in instruction. VanLehn [Van83] and Grosz [Gro77] provide further insight into the constraints and heuristics employed in expert–apprentice dialogues.

It might be hoped that current models of discourse recognition could be used for understanding instruction; that by using schemas and heuristics similar to those in Cawsey's system in a plan recognition system perhaps the process could be reversed to achieve understanding of explanations. However, some fundamental problems arise as evidenced by an inability to explain such discourse phenomena as interleaved subactions. The main responsibility for this falls on the handling of the discourse context. Grosz's discourse processing system treated changes of focus as subtask boundaries and kept track of focus with a global structure, so it fails on two counts: (1) it has no ability to associate an attentional context with an interrupted task; and (2) it makes incorrect assumptions about the relationship between attentional shifts and discourse content. Litman's system accommodated partial context switches through a stacking mechanism, so rather than noting interleaving subtasks her system would perceive a series of interruptions. As well since each interruption would introduce a fresh linguistic context the connection between related portions of a task would be lost. Cawsey's approach suffers from this same flaw.

Another fundamental problem is that of choosing a single hypothesis during plan recognition. This is the issue addressed by Charniak. As he points out Kautz's formalization completely ignores the issue, and outputs all candidates. Kautz's formalism fails as a cognitive model on this count as well as the fact that the minimality constraint that he employs could eliminate the correct hypothesis — cognitively implausible. But the model that Charniak advocates — the evaluation of all hypotheses and selection of the best as the last step — is also not pleasing as a cognitive model of plan recognition. That model fails to explain garden path phenomena. This is important in the instructional setting, as the optimal teaching method under Charniak's model would be the minimum number of moves that would create the correct Bayesian network, but in fact the teacher must beware of creating an initial hypothesis which the student cannot abandon. VanLehn [Van90] and Andreae [And84] present ideas of incremental procedure construction that might provide useful constraints on recognition.

In conclusion, discourse understanding for learning from instruction relies on a number of seemingly disjoint areas of research. Ideas from each area both inspire and constrain ideas in others.

References

[All83] J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[And84] P. M. Andreae. *Justified generalization: acquiring procedures from examples.* PhD thesis, MIT, Department of Engineering and Computer Science, 1984.

[Aus62] J. L. Austin. *How To Do Things With Words.* Oxford University Press, New York, 1962.

[Caw91] Alison Cawsey. Generating interactive explanations. In *AAAI-91 Proceedings, Ninth National Conference on Artificial Intelligence*, pages 86–91. AAAI, AAAI Press/The MIT Press, 1991.

[CG91] Eugene Charniak and Robert Goldman. A probabilistic model of plan recognition. In *AAAI-91*, volume 1, pages 160–165. AAAI, AAAI Press/The MIT Press, 1991.

[CL91] Claire Cardie and Wendy Lehnert. A cognitively plausible approach to understanding complex syntax. In *AAAI-91 Proceedings, Ninth National Conference on Artificial Intelligence*, pages 117–124. AAAI, AAAI Press/The MIT PRESS, 1991.

[CP79] P. Cohen and C. R. Perrault. Elements of a plan–based theory of speech acts. *Cognitive Science*, 3(3):177–212, 1979.

[GN91] Naresh Gupta and Dana Nau. Complexity results for blocks–world planning. In *AAAI-91 Proceedings, Ninth National Conference on Artificial Intelligence*, pages 629–633. AAAI, AAAI Press/The MIT Press, 1991.

[Gri68] H. P. Grice. Utterer's meaning, sentence meaning, and word meaning. *Foundations of Language*, 4, 1968.

[Gro77] B. J. Grosz. The representation and use of focus in a system for understanding dialogues. In *Proceedings of the Fifth IJCAI*, pages 67–76. IJCAI, Morgan Kaufman, 1977.

[Kau90] H. Kautz. A circumscriptive theory of plan recognition. In P. Cohen, J. Morgan, and M. Pollack, editors, *Intentions in Communications*, chapter 6, pages 105–133. The MIT Press, 1990.

[Lit85] D. Litman. *Plan Recognition and Discourse Analysis: an integrated approach for understanding dialogues.* PhD thesis, University of Rochester, Rochester, New York, 1985.

[Lit88] N. Littlestone. Learning quickly when irrelevant attribute abound: a new linear–threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

[McA91] David McAllester. Observations on cognitive judgements. In *AAAI-91*, volume 2, pages 910–914. AAAI, AAAI Press/The MIT Press, 1991.

[SC75] J. McH. Sinclair and R. M. Coulthard. *Towards an Analysis of Discourse: The English Used by Teachers and Pupils.* Oxford University Press, 1975.

[SC91] Fei Song and Robin Cohen. Temporal reasoning during plan recognition. In *AAAI-91*, volume 1, pages 247–252. AAAI, AAAI Press/The MIT Press, 1991.

[Sea69] J. Searle. *Speech Acts: an essay in the philosophy of language.* Cambridge University Press, 1969.

[Val84] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, November 1984.

[Van83] K. VanLehn. Felicity conditions for human skill acquisition: validating an AI-based theory. Interim Report P8300048, XEROX Palo Alto Research Centers, 1983.

[Van90] Kurt VanLehn. *Mind Bugs: The Origins of Procedural Misconceptions.* The MIT Press, Cambridge, Massachusetts, 1990.