Learning Hierarchical Plans by Reading Simple English Narratives

Dustin Smith
MIT Media Lab
Cambridge, MA USA
dsmit@mit.edu

Kenneth Arnold
MIT Media Lab
Cambridge, MA USA
kcarnold@mit.edu

ABSTRACT
We describe an approach for learning a rich plan representation from a parallel corpus of commonsense narratives. Each narrative is an ordered natural language description of the steps required to accomplish common domestic tasks, including "get the mail" and "make a bed", and there are tens to hundreds of differently written narratives for each task. With the goal of learning a single rich plan structure, we 1) convert each narrative from English statements into a sequence of logical predicates, 2) find a global alignment for the sequences, and 3) use the sequences to construct a single underlying plan representation that can be used in language understanding problems. Doing this requires being able to distinguish different ways to accomplish the same goal from missing information, and recognize and compactly represent recurring plan sub-sequences. We describe a simple algorithm that recursively finds graph cycles by applying rules to merge nodes to learn a sequential, parameterized composition (part-of) and abstraction (is-a) plan hierarchy. We hope that these plan representations will help us learn procedural knowledge from increasingly more sophisticated text, where the sub-goals for various actions are not stated.

Author Keywords
Common sense, knowledge acquisition, machine reading, natural language understanding, story knowledge, plan construction, textual entailment

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces—*Natural Language*

INTRODUCTION
How do we get knowledge into computers? Machine reading holds the promise that computers one day could acquire knowledge by learning from text like people do. But the challenges for machine reading go beyond just parsing each sentence. For efficient and interesting communication, human authors leave out familiar information that can be inferred by human readers.

The result is a bootstrapping problem: to learn from a narrative, you must already have some background knowledge. Before children reach pre-school, they can make causal inferences about narratives they hear [13]: even "simple" machine reading tasks require a lot of implicit **common sense** knowledge.

Plan knowledge—sequential descriptions of future first-person actions—has long been recognized as a critical component of understanding narratives and perhaps all text in general.[1] In the late 1970s, Schank and colleagues proposed **scripts** as such a representation [10], but these were painstakingly authored by hand as unwieldy semantic networks.

Another approach is to ask volunteers to contribute knowledge expressed as natural language statements, from which a machine-interpretable structure can be extracted subsequently. By its definition everyone has common sense, so there is no shortage; however, when contributing commonsense, people *still* semantically compress their knowledge. When asked, "What is the consequence of **eating sushi**?" many knowledge contributors tend to reply with droll and improbable events, like "get sick", rather than the common outcomes, "be satiated" or "enjoy the taste." Each volunteer may only contribute one possible narrative in a given situation, but a corpus of redundant narratives permits reconstructing the planning space from which the narratives were generated. The OMICS corpus meets these specifications.

BUILDING PLANS FROM NARRATIVES

OMICS: A corpus of common sense narratives
The Open Mind Common Sense (OMCS) project [12] has collected commonsense knowledge from Internet users since its founding in 1999. Honda Research Institute set up a clone of OMCS, called the Open Mind Indoor Common Sense OMICS project[2], targeted at collecting *indoor* knowledge,

[1]Many cognitive theorists have proposed that **narrative**-like representations are a fundamental representational substrate in human knowledge organization. See [11, 6, 3].

[2]Freely available as a MySQL database from: `http://openmind.hri-us.com`

Submitted for review to IUI 2009.

namely, knowledge about common household items and activities.

One part of OMICS contains collections of short narratives (sequences of simple imperative sentences) about how to accomplish simple indoor domestic activities. This corpus of stories is particularly well-suited for the task of extracting event hierarchies because it:

Is simple to parse. Most event descriptions in the stories have simple phrase structures, making them easy to parse using natural language processing (NLP) tools.

Has a fixed protagonist and goal. The stories have been collected around common tasks (goals) such as "answer the phone", where each step works towards achieving this goal and has a common implicit subject: "[the protagonist needs to] walk up to phone".

Limits the semantic interpretation space. Texts read by literate adults can make use of an *enormous* interpretation space, and a given narrative constitutes just a single path through a large semantic maze. Common indoor task instructions from OMICS present a constrained, more focused interpretation space that has fewer degrees of freedom.

Provides redundant training examples. It is rare to have a resource of multiple of the same story told different ways. In story telling, there is a strong tendency *not* to include the common sense knowledge. Having multiple examples of the same plan provides a more complete representation than any single task description.

Problems with the narrative corpus

Despite its many advantages, this corpus still presents many challenges. Consider the first four of 52 stories about the task: "sweep floor":

A 1 locate broom
A 2 pick up broom
A 3 run broom across floor, gathering dust into a pile
A 4 throw away the dust and dirt

B 1 locate broom
B 2 run broom across floor to gather dust into a pile
B 3 place dust pile in trash can

C 1 take broom
C 2 move broom back on forth
C 3 move garbage towards some location
C 4 pick up garbage and throw away

D 1 get the water
D 2 splash the water on the floor

These examples illustrate several types of problems we must address:

P1. Multiple ways to say the same thing. A4, B3 and C4 presumably refer to the same underlying event (throwing away debris) but do so in different ways.

P2. Temporal abstractions and nested events. Another problem (central to the common sense problem) is that a description of a plan often contains several sub-plans. As an example (more compelling examples can be found in other stories), step 3 of stories A and B could be considered a concise description of the two-step sequence of C2 and C3. Thus we need to identify and group repeating sub-sequences of events.

P3. Global alignment and the problem of context. Each event assumes some underlying context and it is hard for people to articulate where on the contextual umbilical cord the mother (preceding event) ends and the baby (target event) begins. These contributors were presumably articulating different plans from different contexts. For example, the author of story C assumed that the location of the broom was already known, while the authors of stories A and B included the sub-task of "locate broom". Knowledge always requires an implicit context—one could sensibly trace all of the context back to some arbitrarily distant event: before you can sweep the floor, you must "be born", and so on. Additionally, global alignment must deal with the problem of *missing data*.

P4. Descriptions have causal discontinuation. While stories A, B and C describe the process of "sweeping the floor", story D is describing the related chore of "mopping a floor". This is the relation between goals and sub-goals: the nesting allows the specific details of the sub-events to be abstracted away until they share a common parent, *e.g.*, "clean the floor".

IMPLEMENTATION

How do we represent plans?

Temporal Abstraction with Compositional Hierarchies

To accommodate the different temporal granularities of events (P2), events are represented as part-whole relationships that form *compositional hierarchies* [9] using transitive "part of" relationships describing how events are composed of multiple shorter events.

Categorical Abstraction with Taxonomic Hierarchies

There can be multiple ways of accomplishing a goal (P4). When there are multiple mutually exclusive event paths toward achieving a subgoal, we want to represent these alternatives with an *taxonomic hierarchy*, formed from the transitive "is-a" relationship. For example, when you are checking your mailbox for mail, the script that you follow when retrieving mail from your home's mailbox will diverge from the script you will follow when you collect mail from a Post Office; yet, both of these different scripts fall into the more general "get mail" action category.

Sequences with Transitive Closure

When people read stories, the causal structure plays a large role in narrative comprehension, recollection, and summarization [5]. A transitive sequential relationship, stating that one event happens after another, provides a weak ordering for events of the same temporal and categorical level. The plans are already sequential, but together we need to align the sequences between plans to accommodate our first (P1) and third problems (P3).

Generalized Arguments using Background Knowledge
The predicate argument structure we extract from the sentences typically takes the form of a verb-argument pairing. From linguistic "predicate" to logical "predicate", we can re-present the pairing as a *ground atom*, that is, a predicate with its arguments filled in. For example, "open the door" would become OPEN(*door*). By generalizing the arguments, we can detect when a sequence of events, such as ⟨ GRAB(*door*, *knob*), OPEN(*door*), ... ⟩, takes the same entity as argument and create a *abstract symbol* with the argument replaced by a variable, or more general category: ⟨ GRAB(X, *knob*), OPEN(X), ... ⟩. This helps solve (P1).

Example of a learned Plan

We are beginning with a corpus of people's descriptions of everyday activities, described as a sequence of simple sentences. For example, we have 68 stories about the task "answer the phone". From the two narratives, t_1 = ⟨"walk to phone", "grab phone", "say hi"⟩ and t_2 = ⟨"reach in pocket", "grab phone", "say hi"⟩, our goal is to learn a single sequential, parameterized composition and abstraction plan hierarchy. Shown:

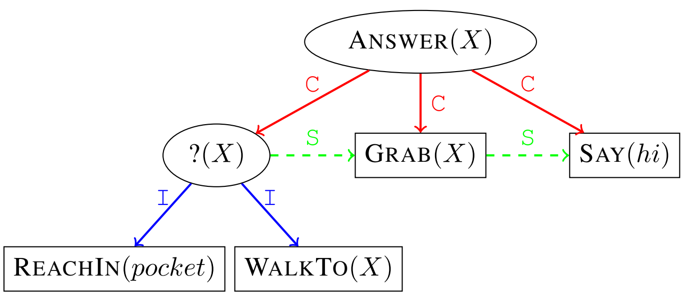


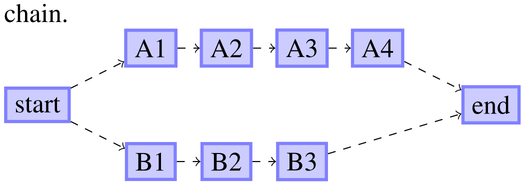
Figure 1. A plan learned from narratives t_1 and t_2, demonstrating compositional abstraction C$(\cdot,\cdot)$, category abstraction I$(\cdot,\cdot)$, and sequential relations S$(\cdot,\cdot)$.

The predicates surrounded by rectangles are leaf nodes that cannot be specialized (have no I children) nor sub-divided (have no C children). The unknown predicate $?(X)$ was introduced as an *is-a* abstraction barrier.

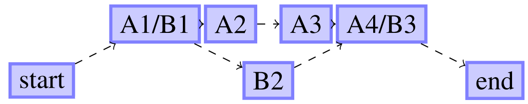
Our approach

1. For each narrative, **convert** the English statements to a sequence of machine interpretable *predicate-argument* forms.

   (a) Add the story to the directed plan graph as a linked chain.



   (b) Then, **align the sequence** to nodes in the graph by joining nodes at certain points with identical predicate-arguments and similar positions.



2. Find all cycles containing a shortest path of length 3, compare inner nodes and

   (a) if they have the same arguments, arrange these into a compositional hierarchy using C$(\cdot,\cdot)$ relations; otherwise,

   (b) represent this causal divergence as an *is-a* I$(\cdot,\cdot)$ abstraction.

   Repeat step 2 until no cycles with 3 node paths exist.

Predicate-Argument Extraction
To extract the predicate-argument structure, we used a modified version of the Link Grammar parser[3] to generate s-expressions of only the predicates and arguments and their nested structure. Next, the predicate-argument structure was transmuted into logical predicates, and we authored several rules to fix common parser mistakes and expand logical connectors. For example `(pred find (arg mop) and (arg bucket))` would be rewritten as two separate ground atoms: FIND(MOP) and FIND(BUCKET). Statistics about the resulting parsed plans can be found in Table 1.

Finding cycles and merging predicate-argument nodes
The result of processing the first two stories for the "get mail" goal is shown below:

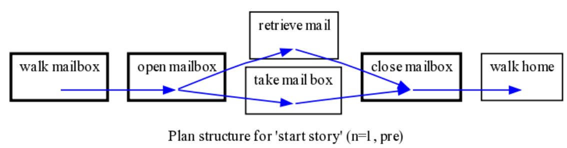


Figure 2. 2 stories, before merge.

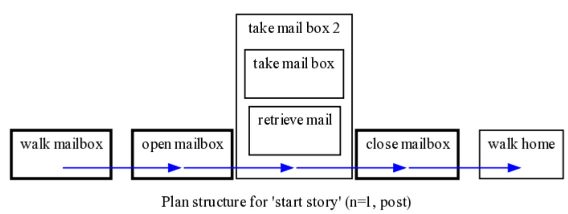


Figure 3. 2 stories, after merge

The two graphs above shows that the two stories are nearly identical, except that the middle action can be described by either "take mail out of box" or "retrieve mail". The graph structure gives evidence that these nodes are functionally equivalent: they are what happens between the same two other steps (opening the mailbox and closing it) and are part of the same sub-goal. Because they share the same arguments, we merge this goal into a single abstracted goal.

Task	PAs	Scripts	PAs/Script
access the internet	100	33	3.030303
answer the phone	63	31	2.032258
check for weather	80	30	2.666667
clean the dishes	133	39	3.410256
close the blinds	51	22	2.318182
close the curtains	64	32	2.000000
cook fish	132	42	3.142857
draw the curtains	76	36	2.111111
feed a pet cat	113	51	2.215686
feed a pet dog	82	27	3.037037
fetch a cold drink	134	47	2.851064
find the time	56	26	2.153846
follow someone around	83	34	2.441176
get mail	79	31	2.548387
get the newspaper	100	41	2.439024

Table 1. Total number of distinct predicate-arguments, count of unique scripts, and mean number of steps per script for the first 15 tasks in the narrative corpus after predicate-argument extraction.

RELATED WORK

Story understand tasks use models that have many components, and often chain many natural language processing tools together. Generally, a smaller sub-problem is approached and the problem solving strategy depends on the target text. Erik Mueller used a script representation and common sense knowledge base to guide inferences in order to answer common questions space and time in a complex narrative [8]. In a more recent project, Mueller treated the story understanding problem as first identifying the protagonist's goals, and then planning according to those goals, using background knowledge and textual knowledge [7]. Chambers and Jurafsky [2] learned unsupervised "event chains" (partially ordered sets of predicate-argument pairs) from a corpus. They used entities (subjects) to identify related predicates from a broad corpus and used a pointwise mutual information on the argument's similarity to recognize equivalent predicate argument structures. These steps would have not worked with our approach because there is no need to find the related predicate (the corpus already aligns similar tasks), and the statements are sequentially ordered.

FUTURE DIRECTIONS

The graph merging approach is limited because of problems with finding a global alignment for all sequences. The problem is exacerbated by the problems of missing data (P3) and textual entailment (P1). We are extending our work to use Logical hidden Markov models (LOHMMs) [4]. LoHMMs extend HMMs, which are effective at characterizing sequential data with hidden states, to include relational (first-order logic) state descriptions. Consequently, the predicates and their arguments can be generalized into abstract categories, the impetus to generalize typically being a compact description of the data. We plan to use an auxiliary semantic knowledge base to allow a predicates' arguments to be generalized (using taxonomic relationships), and tolerance to synecdoche (composition) and synonym. An example of generalizing state descriptions, using only taxonomic relationships in the fully-observable case can be found in [1]. We will use the **narrative cloze** (as described in [2]) to evaluate the system's ability to predict a missing sequence through holding-out some of the stories from the training data. Forthcoming such results will be available from the first author's webpage.

[3] `http://www.link.cs.cmu.edu/link/`

REFERENCES

1. C. Anderson, P. Domingos, and D. Weld. Relational markov models and their application to adaptive web navigation. *Proceedings of the eighth ACM SIGKDD international* ..., Jan 2002.
2. N. Chambers and D. Jurafsky. Unsupervised Learning of Narrative Event Chains. 2008.
3. A. Glenberg, B. Jaworski, M. Rischal, and J. Levin. What brains are for: Action, meaning, and reading comprehension. *Reading Comprehension Strategies: Theories*, Jan 2007.
4. K. Kersting, L. De Raedt, and T. Raiko. Logical hidden markov models. *Journal of Artificial Intelligence Research*, 25:425–456, 2006.
5. J. Lynch, P. V. D. Broek, K. Kremer, P. Kendeou, M. J. White, and E. Lorch. The development of narrative comprehension and its relation to other early reading skills. *Reading Psych.*, 29(4):327–365, Jul 2008.
6. M. Minsky. *The emotion machine*. Simon & Schuster, 2006.
7. E. T. Mueller. Story understanding through multi-representation model construction. *Proceedings of the HLT-NAACL 2003 workshop on Text meaning-Volume 9*, pages 46–53, 2003.
8. E. T. Mueller. Modelling space and time in narratives about restaurants. *Literary and Linguistic Computing*, 22(1):67–84, Nov 2005.
9. K. Pfleger. Learning predictive compositional hierarchies. *Proceedings of the AAAI2000 workshop on New Research* ..., Jan 2000.
10. R. C. Schank and R. P. Abelson. *Scripts, Plans, Goals, and Understanding*. Laurence Erlbaum Associates, Hillsdale, NJ, 1977.
11. R. C. Schank and R. P. Abelson. Knowledge and memory: The real story, 1995.
12. P. Singh, T. Lin, E. T. Mueller, G. Lim, T. Perkins, and W. L. Zhu. Open mind common sense: Knowledge acquisition from the general public. In *On the Move to Meaningful Internet Systems*. Springer-Verlag, 2002.
13. J. Wenner and P. Bauer. Bringing order to the arbitrary: one-to two-year-olds' recall of event sequences. *Infant Behavior and Development*, Jan 1999.