LANGUAGE GENERATION 4
INTRODUCTION TO MULTISENTENCE TEXT GENERATION,
SCHEMAS, AND DISCOURSE RELATIONS

Themes

Multisentence text is not just single-sentence text strung together: producing an understandable and fluent paragraph involves issues of coherence, cohesion, focus shift, and theme development.

How to automate this? The simplest solution is a schema—template generation on the paragraph level. McKeown's thesis. Her need to introduce focus.

Other, more general, approaches to the nature of coherence of texts: theories of Hobbs, DRT, and Grosz and Sidner.

Rhetorical Structure Theory, its relations and its problems. Extensions of relations.

1. The Puzzle of Multisentence Discourse

While any 3-year old can make single sentences, and learns language unconsciously, it takes years to learn to write coherent multisentence text, and after learning it we still cannot describe what makes text coherent.

What is 'coherence'?

Why are there no 'text grammars'? — There is no small set of classes to serve as nonterminal symbols at the multisentence level, and there is no canonical order of these parts.

So what then is text structure? Why can one not simply utter sentences in random order? Some genres are structured, others not.

Current theory: What happens is that text segments have *functions*—they fulfill specified roles that correspond to purposes within the overall message. These functions are often quite recognizable and may even obey some (weak) ordering constraints in some genres. The same sentence may fulfill different functions in different contexts (unlike a word, which has a fixed grammatical category). Some functions: identify, exemplify, contrast, provide-attributes, conclude, provide-reasons, etc.

Consider the texts in Appendix 1. Try to assign a function to each numbered clause. These examples are taken from McKeown's thesis (1985).

2. Schemas Implemented in the Text System

In the early 1980s, McKeown built the TEXT system to generate paragraphs from database info. After analyzing a lot of text, she identified four stereotypical text structures, which she called 'schemas':
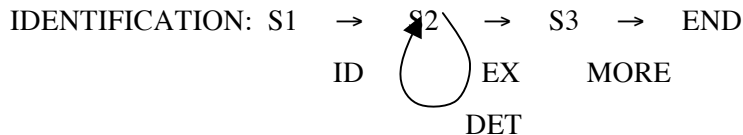
- Identification
- Constituency

- Attributive
- Compare&Contrast.

Each schema was a paragraph-length template controlling the order not of words but of clauses, each clause with a particular function. She defined the following rhetorical predicates (functions):

Adversative, Amplification, Analogy, Answer, Attributive, Cause-effect, Comparison, Constituency, Contrast, Evidence, Explanation, Generalization, Identification, Inference, Particular-illustration, Positing Question; Problem, Renaming, Representative, Restriction

She implemented each schema as an ATN:

IDENTIFICATION: S1 → S2 → S3 → END

ID ( ) EX MORE

DET

ID: generate: Identification (i.e., token class + attribute/function)
DET: generate one of: Analogy / Constituency / Attributive / Renaming
EX: generate one of: Particular-illustration / Evidence
MORE: generate one of: Amplification / Analogy / Attributive / Particular-illustration / Evidence

Naturally, each rhetorical predicate had to match some info in the database:

- Identification matches the :name slot,
- Attributive matches :size, :weight, etc. slots,

and so on.

Initial implementation: Once the basic info had been extracted from the database and collected into a list of packets resembling case frames, the packets were sent to the sentence realizer, each one producing a clause.

McKeown's 4 schemas

/ = OR
* = 0 or more times
+ = 1 or more times
{} = optional

IDENTIFICATION SCHEMA

Identification (= class + attribute/function)	identify the object
{Analogy / Constituency / Attributive / Renaming}*	make an analogy or list its parts or attributes or other name
Particular-illustration / Evidence +	give a specific example or evidence for it
{Amplification / Analogy / Attributive}	give more details or an analogy or give attributes
{Particular-illustration / Evidence}	give a specific example or evidence for it

COMPARE & CONTRAST SCHEMA

Positing / Attributive (about topic ~A)
{Attributive (A) / Particular-illustration (A) / Evidence (A) /

Amplification (A) / Inference (A) / Explanation (A)}
  {Comparison (A and ~A) / Explanation (A and ~A) / Generalization (A and ~A) /
   Inference (A and ~A)} +

CONSTITUENCY SCHEMA
  Consituency
  Cause-effect* / Attributive* / {Depth-identification / Depth-attributive
   {Particular-illustration / Evidence}
   {Comparison; Analogy} } +
  {Amplification / Explanation / Attributive / Analogy}

ATTRIBUTIVE SCHEMA
  Attributive
  {Amplification; Restriction}
  Particular-illustration *
  {Representative}
  {Question;Problem Answer}
  {Comparison;Contrast Adversative}
  Amplification / Explanation / Inference / Comparison}

Using these definitions, try to decide which schema applies to each text in Appendix 1.
Example text 2 (Eltville): 1. Identification (class &attribute) 2. Attributive, 3. Amplification, 4.
Particular illustration.
Example text 1 (Movies): 1. Positing ~A, 2. Attributive of A, 3. Evidence for A, 4.
Comparison/explanation A and ~A, 5. Inference A and ~A, 6. Comparison/explanation A and ~A.
Example text 3 (torpedoes): 1. Constituency, 2. Depth-identification (attributive), 3. Comparison,
4. Depth-identification (attributive).
Example text 4 (violence): 1. Attributive, 2. Amplification, 3. Particular-illustration, 4.
Representative, 5. Amplification/explanation.

Some Problems and Further Work

Question 1:

The rhetorical predicates were defined by McKeown in terms of database relations and types.
But what happens when you use a different database? What is the general definition for
something like Exemplification or Attributive? What are the implications of the difference
between Attributive (as simple slot-filler) and something 'large' like Explanation? We discuss
this in the next lecture, on Rhetorical Structure Theory.

Question 2:

Why do the schemas have the structure they have? Do all texts conform to schemas? How do
those for which no obvious schema can be found work? Are they truly free-form? What is the
nature of communicative role or intent? How is it that we understand the role of each portion to
be what it is, even when there is no clear schema to guide us?

Even for genres with relatively clear conventional structure, such as encyclopedia articles describing physical objects, one finds differences in a single schema. Paris, in her PhD work, identified a large pragmatic difference in this genre: children's encyclopedias focused on function while adults' articles focused on form. Paris thus developed a much more refined set of IDENTIFICATION schemas with choice criteria based also on pragmatic factors (Paris, 1987). How should one handle this?

3. The Problem of 'Focus'

The straightforward implementation gave functional but clumsy text:

> Compare&Contrast "destroyer" and "bomb":

> "A destroyer is a surface ship with a draft between 15 and 222. A ship is a vehicle. A bomb is a free falling projectile that has a surface target location. A free falling projectile is a lethal destructive device. The bomb and the destroyer, therefore, are very different kinds of entities."

You can more or less directly see the schema structure operating here. You can also feel how clumsy this text is. It can be cleaned up somewhat by better reference planning (pronouns such as "it") and aggregation (joining together two clauses and dropping out the shared parts). This kind of 'smoothing' is called Sentence (micro-) Planning. It operates after Text (macro-) Planning has chosen the content, and before realization turns the content into sentences.

But another problem presents itself—at some points, McKeown's system could say more than one thing (more than one alternative in the schema were matched and retrieved from the database). For example, the Analogy can be made in both steps 2 and 4 of the Identification schema. Where does it fit best? That depends on which was said before, and what the analogy is. How can one determine this?

Consider the following example from McKeown:

Version 1:

1.1) A hobie cat is a brand of catamaran,
1.2) which is a kind of sailboat.
1.3) Catamarans have sails and a mast like other sailboats,
1.4) but they have two hulls instead of one.
1.5) That's a catamaran there.
1.6) Hobie cats have a canvas cockpit connecting the two pontoons and one or two sails.
1.7) The 16 ft hobie cat has a main and a jib and the 14 ft. hobie cat has only a main.

Version 2:

2.1) A hobie cat is a brand of catamaran,
2.2) which is a kind of sailboat.
2.3) The 16 ft hobie cat has a main and a jib and the 14 ft. hobie cat has only a main.
2.4) Hobie cats have a canvas cockpit connecting the two pontoons and one or two sails.
2.5) Catamarans have sails and a mast like other sailboats,
2.6) but they have two hulls instead of one.

2.7) That's a catamaran there.

Which version is preferable? Why?

McKeown argues that at any point in the text there is a 'focus space'—a set of entities that are immediately available from the items in the current clause—and that one has to use them in the next clause, unless one performs an explicit focus shift. After a shift, you are in a new focus space, and can hence talk about the things in that space. This idea is an elaboration of the PhD work of Candy Sidner and relates to a later PhD thesis by Kathy McCoy.

McKeown defines a Focus Space with respect to the knowledge representation system more or less as follows: any node that is directly connected (= semantic distance 1) to the node in question is in its Focus Space. In the above example, Focus Space 1 includes sailboats in general, with all their properties (sails and hulls and so on). Focus Space 2 centers on the hobie cat in particular, with its two hulls and the various numbers of sails for the different lengths of hobie cats. According to McKeown, Version 1 is ok because you first finish the material in Focus Space 1, which has been set up by 1.2, before moving on to Focus Space 2. Version 2, however, is not because you jump immediately after setting up Focus Space 1 (in 2.2) to Focus Space 2 (in 2.3 and 2.4) and then return *back* to Focus Space 1 (in 2.5 and 2.6).

So she implements focus in TEXT as follows:
- a register called Current Focus (CF) that carries the data item in current focus,
- a list, the potential focus list (PFL), of items to go to
- a stack of all foci previously mentioned (that you can pop back to)

with the algorithm:

1. change focus to something in PFL if possible:
   CF(new clause) ← element(PFL(previous clause))

2. else maintain focus if possible:
   CF(new clause) ← CF(previous clause)

3. else return to previous topic:
   CF(new clause) ← element(focus-stack)

Combined with the ATNs above, this mechanism has the effect of using up all the data items semantically 'close' to the most recent topic before moving 'further'.

To get a better idea of 'focus', read the variations of an example attached to these notes and see which one(s) you prefer. Explain why. More recent work by computational linguists like Joshi, Grosz, and others have elaborated the idea of Focus Space and related notions into something called Centering Theory (Walker et al., 1999).

See the text variations in Appendix 2, and decide which ones you like, and why.

4. Computation-oriented Theories of Discourse Structure

The schemas work fine for very stereotypical paragraphs. But most text is not so fixed. One would like a general theory of the structure of discourse (multi-sentence text). Many people have tried to build one, but it turns out to be harder to specify than sentence grammar.

The ancient Greeks studied rhetoric, to learn what made some speeches good. Already there people recognized that different parts of a text had distinct functions. In his *Rhetoric* Aristotle

proposed a few major functions of segments of a speech—the introduction, the main argument, etc. Through the ages, Text Linguistics has been a big field of study.

In Computational Linguistics, work on discourse structure started in the late 1970s.

In text parsing, the main players were the following.

Grosz and Sidner: They consider discourse revolving around a task performed by two collaborating agents (assembling a pump, for example). Here the discourse structure closely parallels the task structure. Their theory of discourse, which is often cited, has the following main points:

- discourse breaks into **segments**
- each discourse segment has a distinct purpose or **Intention** (usually very close to a (sub)purpose of the task)
- the discourse segment units can be arranged into a tree structure called the **Intentional Structure**, using ID/LP (immediate dominance / linear precedence) links. Each node represents a discourse segment, and the tree's links indicate the segment nesting and internal order (i.e., there is no semantics to the links themselves)
- parallel to the intentional structure, one has to record also what the agents are talking about (focusing on), similar to the work on focus on McKeown and McCoy. This is done using the **Attentional Structure**, which is implemented as a stack.
- Together, the Intentional Structure and its parallel Attentional Structure describe what the agents *say*. These structures mirror the agents' Plan Structure, which governs what they *do*. The implication (although not stated) is that these two structures are not isomorphic.

Work in the 1990s (Lochbaum and Grosz) elaborated this model to look at shared plans and collaborative multi-agent planning.

While Grosz and Sidner's work is very often cited, it has not given rise to many actual computational systems. This is probably because there is too little detail in the theory, compared to RST or DRT. What types of intentions are there? How does each intention constraint what its segment's contents are? How is the order to segments controlled, and why? What kind of information can be transferred across segment boundaries, and what not? (For example, many people claim that pronouns "he", "it", etc., respect major discourse boundaries: when you cross over one, you have to re-introduce the person or object. How does the theory work in this regard?)

Hobbs: Hobbs rejects the idea of a discourse structure distinct from semantics. He views discourse as imply the network of meanings when you put together sentence meanings, properly represented by a semantic network in which individual clauses and sentences are simply embedded portions. In some sense, one could say that there is little difference between a sentence-sized case frame and the paragraph-sized matrix of sentence frames; they all exist in a highly connected graph structure. He identifies a set of relations between representational elements that are typically used between (and not within) separate clauses in a text: Cause, Enablement, etc. Hobbs does not pay attention to the problems of Focus (or other discourse phenomena such as Theme and Rheme).

Most interestingly, Hobbs provides an initial formalization of the logical import of these relations using first-order predicate calculus. See Appendix 3 for an excerpt of one of Hobbs's papers.

6

Polanyi: Polanyi's discourse model is more logically sophisticated than that of Grosz and Sidner. Also, it includes the discourse segmentation ideas that Hobbs lacks. But her ideas have never really been taken up by others.

Discourse Representation Theory (DRT): Kamp, and later Kamp and Asher, devoted a great deal of time over three decades to an extremely formal, logic-based account of how groups of clauses work together to produce interpretations. The approach resembles Richard Montague's formal logic approach to sentence-level semantics. Despite the somewhat ambitious name, DRT for about two decades was mainly a theory of quantifier scoping and reference tracking over multiple clauses. For example:

1. *Pedro looks after many people's donkeys every day.*
2. *He loves donkeys but sometimes gets irritated by them.*
3. *Pedro can't touch the others, but he often beats every donkey he owns.*
4a. *They are much better behaved.* | 4b. *They are generally irritating animals.*

The second clause of sentence 3 most people will express in first order logic as

$$\forall(x) \ (Donkey(x) \ \& \ Belongs\text{-}to(Pedro,x) \ \rightarrow \ Beats(Pedro,x))$$

The problem is how to interpret sentences 4a and 4b. Do the "they"s refer to just Pedro's donkeys or all of them? That is, how far does the quantification over x hold—just in sentence 3, second clause, or into sentences 4a and b? Why does it seem more likely to hold also for 4a than 4b—what is it in the sentences that sways our preference? Before Kamp's work, there had been no attempt in Montague semantics to deal with how to know how "they" refers across sentences

In DRT, little 'boxes' called DRs are created that contain the clauses and bind the referents to their logical identifier, thereby delimiting their scope. These DRs can be nested in the obvious way. Eventually, after two decades, the result is rather similar to Grosz and Sidner's intuitive idea of discourse segments, where now each segment carries not only a purpose but also a list of the referents known inside it, and their quantities. A later addition to DRT, by Asher especially, incorporates discourse relations, to which we turn next.

Rhetorical Structure Theory (RST): Developed in the 1980s, Mann and Thompson's influential collection of discourse structure relations, anchored on discourse relator words such as "so", "then", as well as other structures. Coherent texts are representable as tree structures in which the relators are nested correctly to achieve the overall rhetorical effect. See the next lecture.

Penn Discourse Treebank (PDTB): Joshi, Webber, Nenkova, and others hold that discourse relator words act like two-place 'verbs' between sentences. They created an annotated text corpus of these 'discourse trees'.

5. Parsing Discourse Structure

The earliest serious computational work on discourse parsing is Marcu's 1997 PhD dissertation, followed by similar work by Corston-Oliver. There have been some recent attempts by others to extend it.

Optional Readings

Schemas and Focus

McKeown, K.R. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Cambridge University Press.

Paris, C.L. 1987. The Use of Explicit User Models in Text Generation: Tailoring to a User's Level of Expertise. Ph.D. dissertation, Columbia University.

Walker, M, A.K. Joshi, and E. Prince. 1999. *Centering Theory in Discourse*. Clarendon Press.

Discourse theories

Asher, N.

Grosz, B.J. and Sidner, C.L. 1986. Attention, Intentions, and the Structure of Discourse. *Computational Linguistics* 12, 175–204.

Hobbs, J.R. 1979. Coherence and Coreference. *Cognitive Science* 3, 67–90.

Hobbs, J.R. 1993. From *Burning Issues in Discourse*, edited by E.H. Hovy and D.R. Scott.

Kamp, H. 1981. A Theory of Truth and Semantic Representation. In J.A.G. Groenendijk, Th.M.V. Janssen and M.B.J. Stokhof (eds), *Formal Methods in the Study of Language*, Amsterdam: Mathematisch Centrum Tracts 135, 25–37.

RST

Mann, W.C. and Thompson, S.A. 1988. Rhetorical Structure Theory: Toward a Functional Theory of Text Organization. *Text* 8, 243–281. Also available as USC/Information Sciences Institute Research Report RR-87-190.

Moore, J.D. and M.E. Pollack. 1992. A Problem for RST: The Need for Multi-Level Discourse Analysis. Squib in *Computational Linguistics* 18(4).

Other Systems of Relations

Hovy, E.H., J. Lavid, E. Maier, V. Mittal, and C.L. Paris. 1992. Employing Knowledge Resources in a New Text Planning Architecture. In R. Dale, E. Hovy, D. Rösner, and O. Stock (eds) *Aspects of Automated Natural Language Generation*. Heidelberg, Germany: Springer Verlag Lecture Notes in AI 587, (57–72).

Joshi, A. 2008. Towards Discourse Meaning. In *Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs*. Springer Lecture Notes Vol 5320.

Sanders, T.J.M., W.P.M. Spooren, and L.G.M. Noordman. 1992. Towards a taxonomy of coherence relations. *Discourse Processes* 15:1–35.

Corpora Annotated with Relations

Carlson, L., Marcu, D., Okurowski, M.E. 2001. Building a discourse-tagged corpus in the framework of rhetorical structure theory. *Proceedings of the 2nd SIGDIAL Workshop on Discourse and Dialogu*e.

Joshi, A. 2008. Penn Discourse Treebank. http://www.seas.upenn.edu/~pdtb/

Parsing Discourse Structure

Marcu, D. 2000. The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics* 26, 395–448.

Appendix 1

McKeown example texts
Try to identify the discourse function of each numbered clause in the texts below.

Compare the patterns you find for these four paragraphs.

<u>Text 1</u>

Eltville, (Germany) 1) An important wine village of the Rheingau region. 2) The vineyards make wines that are emphatically of the Rheingau style, 3) with a considerable weight for a white wine. 4) Taubenberg, Sonnenberg, and Langenstück are among vineyards of note. (Paterson 80)

1)	2)
3)	4)

<u>Text 2</u>

Steam and electric torpedoes. 1) Modern torpedoes are of 2 general types. 2) Steam-propelled models have speeds of 27 to 45 knots and ranges of 4000 to 25,000 yds (4,367–27,350 meters). 3) The electric powered models are similar 4) but do not leave the telltale wake created by the exhaust of a steam torpedo. (Encyclopedia 76)

1)	2)
3)	4)

<u>Text 3</u>

This book, being about work, is, by its very nature, about violence — 2) to the spirit as well as to the body. 3) It is about ulcers as well as about accidents, about shouting matches as well as fistfights, about nervous breakdowns as well as kicking the dog around. 4) It is, above all (or beneath all), about daily humiliations. 5) To survive the day is triumph enough for the walking wounded among the great many of us. (Terkel 72)

1)	2)
3)	4)
5)	

<u>Text 4</u>

1) Movies set up these glamorized occupations. 2) When people find they are waitresses, they feel degraded. 3) No kid says "I want to be a waiter, I want to run a cleaning establishment". 4) There is a tendency in movies to degrade people if they don't have white-collar professions. 5) So, people form a low self-image of themselves, 6) because their lives never match the way Americans live—on the screen. (Terkel 72)

1)	2)
3)	4)
5)	6)

Appendix 2

Discourse structure and logical inference

An extract from Hobbs (1993)

Abductive inference can be defined as "inference to find the best explanation". The process of interpreting sentences in discourse can be viewed as the process of providing the best explanation of why the sentences would be true. This insight can be cashed out procedurally in terms of theorem-proving technology as follows:

To interpret a sentence:

(1) Prove the logical form of the sentence,
    including the constraints predicates impose on their arguments,
    allowing for semantic coercions (e.g., "Washington announced that…" →
        "a spokesperson for the Government in Washington announced that…"),
    merging redundancies where possible,
    making assumptions where necessary.

In a discourse situation, the speaker and hearer both have their sets of private beliefs, and there is a large overlapping set of mutual beliefs. An utterance spans the boundary between mutual belief and the speaker's private beliefs. It is a bid to extend the area of mutual belief to include some private beliefs of the speaker's. It is anchored referentially in mutual belief, and where we succeed in proving the logical form and the constraints, we are recognizing this referential anchor. This is the given information, the definite, the presupposed. Where it is necessary to make assumptions, the information comes from the speaker's private beliefs, and hence is the new information, the indefinite, the asserted. Merging redundancies is a way of getting a minimal, and hence a best, interpretation.

Choosing the best or minimal interpretation relies on an algorithm for weighted abduction that levies variable costs for assumptions and for length of proof and reduces costs when redundancies are recognized.

Hobbs treats an **example from Grosz and Sidner** as follows.

Let us analyze an example from a set of dialogues collected by Grosz (1977) between an expert and an apprentice engaged in fixing an air compressor. They are in different rooms, communicating by terminals. The apprentice A is doing the actual repairs, after receiving instructions from the expert B. At one point, the following exchange takes place:

B: Tighten the bolt with a ratchet wrench.
A: What's a ratchet wrench?
B: It's between the wheel puller and the box wrenches.

A seems to be asking for a definition of a ratchet wrench. But that is not what B gives her. He does not say

A ratchet wrench is a wrench with a pawl, or hinged catch, that engages the sloping teeth of a gear, permitting motion in one direction only.

Instead he tells her where it is.

According to a plausible analysis, B has interpreted A's utterance by relating it to A's overall plan. B knows that A wants to use the ratchet wrench. To use a ratchet wrench, you have to know where it is.

To know where it is, you have to know what it is. B responds to A's question, not by answering it directly, but by answering to a higher goal in A's presumed overall plan, by telling A where it is.

B has therefore recognized the relationship between A's utterance and her overall plan. I will give two accounts of how this recognition could have taken place. The first account is informational. It is derived in the process of proving the logical form. The second account is intentional and subsumes the first. It is derived in the process of explaining, or proving abductively, the fact that A's utterance occurred.

To describe the **Information portion of this solution** we will need two axioms encoding the planning process:

(3) $(\forall a, e_0, e_1)$ $goal(a, e_1)$ $\wedge$ $enable(e_0, e_1)$ $\supset$ $goal(a, e_0)$

or if an agent a has e_1 as a goal and e_0 enables, or is a prerequisite for, e_1, then a has e_0 as a goal as well.

(4) $(\forall a, e_0, e_1)$ $goal(a, e_1)$ $\wedge$ $cause(e_0, e_1)$ $\wedge$ $etc_1(a, e_0, e_1)$
$$\supset goal(a, e_0)$$

or if an agent a has e_1 as a goal and e_0 causes, or is one way to accomplish, e_1, then a may have e_0 as a goal as well. The etc_1 literal encodes the uncertainty as to whether e_0 will be chosen as the way to bring about e_1 rather than some other action that causes e_1.

In terms of STRIPS operators (Fikes and Nilsson, 1971), the first axiom says that prerequisites for an action must be satisfied, while the second axiom says essentially that to achieve a goal, an operator needs to be chosen and its body (e_0) needs to be executed.

Next we need two domain axioms of a rather general character.

(5) $(\forall e_2, a, x)$ $use'(e_2, a, x)$ $\supset$ $(\exists e_3, e_4, y) enable(e_3, e_2)$ $\wedge$ $know'(e_3, a, e_4)$ $\wedge$ $at'(e_4, x, y)$

or an agent a's use e_2 of a thing x has as a prerequisite a's knowing e_3 the fact e_4 that x is at someplace y. To use something, you have to know where it is.

(6) $(\forall e_3, a, e_4, x, y)$ $know'(e_3, a, e_4)$ $\wedge$ $at'(e_4, x, y)$
$$\supset (\exists e_5, e_6) enable(e_5, e_3) \wedge know'(e_5, a, e_6) \wedge$$
$wh'(e_6, x)$

or an agent a's knowing e_3 the fact e_4 that a thing x is at someplace y has as a prerequisite a's knowing e_5 what x is (e_6). To know where something is, you have to know what it is. We dodge the complex problem of specifying what constitutes knowing what something is by encoding it in the predicate wh, which represents the relevant context-dependent essential property.

Let us suppose that the logical form of

What's a ratchet wrench?

is

(7) $(\exists a, e_5, e_6)$ $goal(a, e_5)$ $\wedge$ $know'(e_5, a, e_6)$ $\wedge$ $wh'(e_6, RW)$

That is, the speaker a has the goal e_5 of knowing the essential property e_6 of the ratchet wrench RW. Most of this logical form comes, of course, from our recognition that the utterance is a question.

Suppose also that in B's knowledge of the context is the following fact:

(8) $goal(A,E_2) \wedge use'(E_2,A,RW)$

That is, the apprentice A has the goal E_2 of using the ratchet wrench RW.

The proof of the logical form (5) follows from axioms (3) through (6) together with fact (8), as indicated in Figure 2. Axiom (3) is used twice, first in conjunction with axiom (6) and then with axiom (5), to move up the planning tree. The apprentice wants to know what a ratchet wrench is because she wants to know where it is, and she wants to know where it is because she wants to use it. The proof then bottoms out in fact (8).
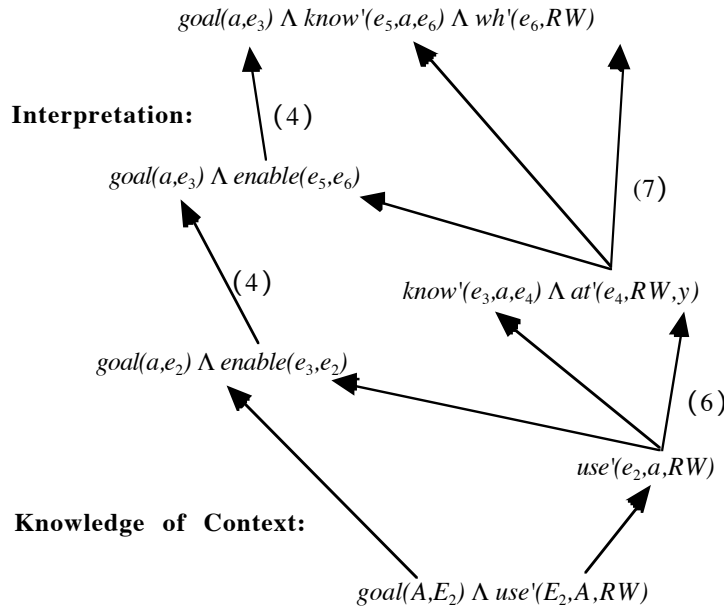
Logical Form:

$$goal(a,e_3) \wedge know'(e_5,a,e_6) \wedge wh'(e_6,RW)$$

Interpretation: (4)

$$goal(a,e_3) \wedge enable(e_5,e_6)$$

(7)

(4)

$$know'(e_3,a,e_4) \wedge at'(e_4,RW,y)$$

$$goal(a,e_2) \wedge enable(e_3,e_2)$$

(6)

$$use'(e_2,a,RW)$$

Knowledge of Context:

$$goal(A,E_2) \wedge use'(E_2,A,RW)$$

Figure 2. Informational interpretation of "What's a ratchet wrench?"

To summarize, if we take the logical form of a question to be the expression of a desire to know something, then the proof of that logical form often involves the recognition of the ultimate aims of the speaker in asking it.

To describe the **Intentional portion of this solution** we proceed as follows. According to the Informational Perspective, it is the logical form of the utterance that needs to be explained, or proven abductively. We will now take a broader view in which it is the occurrence of an event in the world that has to be explained. It is not the content of the utterance that we have to explain, but rather the very fact that the utterance occurred. Frequently, the best explanation of an event is that it is an intentional action on the part of some agent, that is, it is an action in the service of some goal. This is especially true of utterances— they are generally intentional acts. Thus, we will be interpreting the utterance from an Intentional Perspective. We will ask why the speaker said what she did. We will see how this in turn encompasses the Informational Perspective.

We need several more axioms. First we need some axioms about speaking.

$$(\forall e_7,a,b,e_8) \; say'(e_7,a,b,e_8) \supset (\exists e_9) \; cause(e_7,e_9) \wedge know'(e_9,b,e_8)$$

That is, if e_7 is a's saying e_8 to b, then that will cause the condition e_9 of b's knowing e_8. Saying causes knowing. The next axiom is the converse of this.

(9) $(\forall e_k,y,e) \; know'(e_k,y,e) \wedge etc_2(e_k,y,e)$
$$\supset (\exists e_s,x) \; cause(e_s,e_k) \wedge say'(e_s,x,y,e)$$

That is, if e_k is y's knowing the fact e, then it may be (etc_2) that this knowing was caused by the event e_s of x's saying e to y. Knowing is sometimes caused by saying. In the interpretation of the utterance we need only the second of these axioms.

Next we need some axioms (or axiom schemas) of cooperation.

$$(\forall e_5,e_8,e_9,e_{10},a,b) \quad know'(e_9,b,e_8) \quad \wedge \quad goal'(e_8,a,e_5) \quad \wedge \; cause(e_{10},e_5) \quad \wedge$$
$$p'(e_{10},b)$$
$$\wedge \; etc_3(e_5,e_8,e_9,e_{10},a,b) \qquad \supset cause(e_9,e_{10})$$

That is, if e_9 is b's knowing the fact e_8 that a has goal e_5 and there is some action e_{10} by b doing p that causes e_5, then it may be (etc_3) that that knowing will cause e_{10} to actually occur. If I know your goals, I may help with them[1].

The next axiom schema is the converse, being attribution of cooperation.

(10) $(\forall e_5,e_{10},b) \; p'(e_{10},b) \wedge cause(e_{10},e_5) \wedge etc_4(e_5,e_{10},b)$
$$\supset (\exists e_8,e_9,a) \; cause(e_9,e_{10}) \wedge know'(e_9,b,e_8) \wedge goal'(e_8,a,e_5)$$

That is, if an action e_{10} by b occurs, where e_{10} can cause e_5, then it may be (etc_4) that it was caused by the condition e_9 of b's knowing the fact e_8 that a has the goal e_5. Sometimes I do things because I know it will help you. In the example we will only need the axiom in this direction.

Finally, an axiom schema that says that people do what they want to do.

(11) $(\forall a,e_7) \; goal(a,e_7) \wedge p'(e_7,a) \wedge etc_5(a,e_7) \supset Rexists(e_7)$

That is, if a has as a goal some action e_7 that a can perform, then it could be (etc_5) that e_7 will actually occur. This axiom, used in backward chaining, allows us to attribute intention to events.

Now the problem we set for ourselves is not to prove the logical form of the utterance, but rather to explain, or prove abductively, the occurrence of an utterance with that particular content. We need to prove

$(\exists e_7,a,b,e_8,e_5,e_6) \; Rexists(e_7)$
$$\wedge \; say'(e_7,a,b,e_8) \quad \wedge \quad goal'(e_8,a,e_5) \quad \wedge \; know'(e_5,a,e_6) \quad \wedge$$
$$wh'(e_6,RW)$$

That is, we need to explain the existence in the real world of the event e_7 of someone a saying to someone b the proposition e_8 that a has the goal e_5 of knowing the essential property e_6 of a ratchet wrench.

The proof of this is illustrated in Figure 3. The boxes around the "et cetera" literals indicate that they have to be assumed. By axiom (11) we attribute intention to explain the occurrence of the utterance act e_7; it's not like a sneeze. Using axiom (4), we hypothesize that this intention or goal is a subgoal of some

---

[1] More properly, where I have $p'(e_{10},b)$ I should have $agent(b,e_{10})$, together with a set of axioms of the form $(\forall e,x) \; p'(e,x,...) \supset agent(x,e)$.

other goal e_9. Using axiom (9), we hypothesize that this other goal is b's knowing the content e_8 of the utterance. A uttered the sentence so that B would know its content. Using axiom (4) again, we hypothesize that e_9 is a subgoal of some other goal e_{10}, and using axiom (10) we hypothesize that e_{10} is b's saying e_6 to a. A told B A's goal so that B would satisfy it. Using axiom (4) and (9) again, we hypothesize that e_{10} is a subgoal of e_5, which is a's knowing e_6, the essential property of a ratchet wrench. A wants B to tell her what a ratchet wrench is so she will know it.
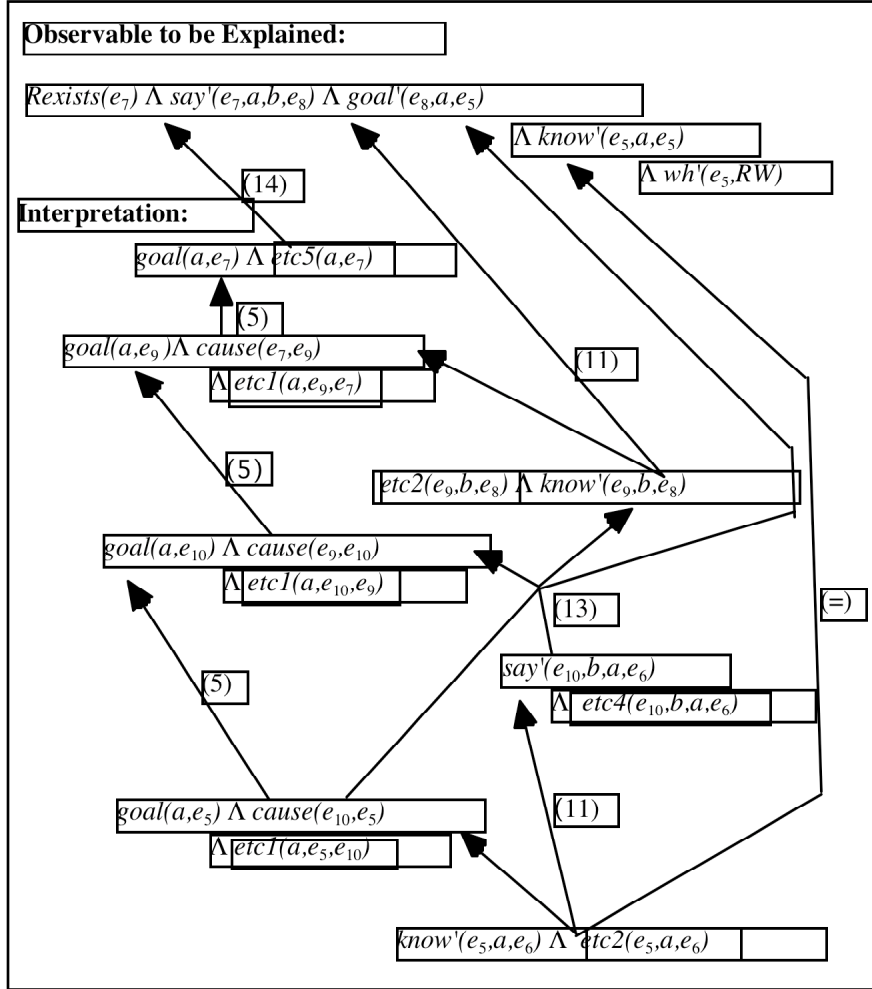

Figure 3. Intentional Interpretation of "What's a ratchet wrench?"

The desired causal chain is this: A tells B she wants to know what a ratchet wrench is, so B will know that she wants to know what a ratchet wrench is, so B will tell her what a ratchet wrench is, so she will know what a ratchet wrench is. Causal chains are reversed in planning; if X causes Y, then our wanting Y causes us to want X. Hence, the causal chain is found by following the arrows in the diagram in the reverse direction.

At this point all that remains to prove is

$(\exists a,e_5,e_6)$ $goal(a,e_5)$ $\wedge$ $know'(e_5,a,e_6)$ $\wedge$ $wh'(e_6,RW)$

But this is exactly the logical form whose proof is illustrated in Figure 2. We have reduced the problem of explaining the occurrence of an utterance to the problem of discovering its intention, and then reduced that to the problem of explaining the content of the utterance. Interpetation from the Intentional Perspective includes as a subpart the interpretation of the utterance from the Informational Perspective.

Appendix 3

Focus Games

OPTION A

Perhaps nobody counts more in this town than the pocket-less employees who work at a nondescript building in Downtown Los Angeles. This is no nickel-and-dime operation.

The facility has no identifying signs but plenty of armed guards and cameras.

They handle the money dropped into Metropolitan Transit Authority bus fare boxes.

A sign on the wall reads: "A reward of up to $5,000 will be paid to any RTD employee who furnishes information resulting in the arrest and conviction of anyone who steals or assists in the theft of RTD property."

When they report to work, security dictates that employees change into uniforms without pockets. Inside the MTA's counting house, 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines: pennies, nickels, dimes, quarters, tokens and on occasion, francs, pesos and Chuck E Cheese's tokens.

The precise location of the counting house the agency keeps secret for security reasons.

The facility processes about $125 million a year in dollars and cents. To enter MTA's cash-counting operation, you must pass through three locked doors.

When they report to work, security dictates that employees change into uniforms without pockets.

LA Times, Metro Section, Monday Sept 19, 1994.

OPTION B

Perhaps nobody counts more in this town than the pocket-less employees who work at a nondescript building in Downtown Los Angeles.

They handle the money dropped into Metropolitan Transit Authority bus fare boxes.

Inside the MTA's counting house, whose precise location the agency keeps secret for security reasons, 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines: pennies, nickels, dimes, quarters, tokens and on occasion, francs, pesos and Chuck E Cheese's tokens.

This is no nickel-and-dime operation. The facility — which has no identifying signs but plenty of armed guards and cameras — processes about $125 million a year in dollars and cents.

To enter MTA's cash-counting operation, you must pass through three locked doors. A sign on the wall reads: "A reward of up to $5,000 will be paid to any RTD employee who furnishes information resulting in the arrest and conviction of anyone who steals or assists in the theft of RTD property."

When they report to work, security dictates that employees change into uniforms without pockets.

LA Times, Metro Section, Monday Sept 19, 1994.

OPTION C

When they report to work, security dictates that employees change into uniforms without pockets. This is no nickel-and-dime operation.

To enter MTA's cash-counting operation, you must pass through three locked doors. A sign on the wall reads: "A reward of up to $5,000 will be paid to any RTD employee who furnishes information resulting in the arrest and conviction of anyone who steals or assists in the theft of RTD property."

Perhaps nobody counts more in this town than the pocket-less employees who work at a nondescript building in Downtown Los Angeles.

They handle the money dropped into Metropolitan Transit Authority bus fare boxes. The facility — which has no identifying signs but plenty of armed guards and cameras — processes about $125 million a year in dollars and cents.

Inside the MTA's counting house, whose precise location the agency keeps secret for security reasons, 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines: pennies, nickels, dimes, quarters, tokens and on occasion, francs, pesos and Chuck E Cheese's tokens.

LA Times, Metro Section, Monday Sept 19, 1994.

OPTION D

Perhaps nobody counts more in this town than the pocket-less employees who work at a nondescript building in Downtown Los Angeles.

They handle the money dropped into Metropolitan Transit Authority bus fare boxes.

Inside the MTA's counting house, whose precise location the agency keeps secret for security reasons, 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines: pennies, nickels, dimes, quarters, tokens and on occasion, francs, pesos and Chuck E Cheese's tokens.

...

OPTION E

Perhaps nobody counts more in this town than the pocket-less employees who work at a nondescript building in Downtown Los Angeles.

The money dropped into Metropolitan Transit Authority bus fare boxes is handled by them.

Inside the MTA's counting house, whose precise location the agency keeps secret for security reasons, 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines: pennies, nickels, dimes, quarters, tokens and on occasion, francs, pesos and Chuck E Cheese's tokens.

OPTION F

Inside the MTA's counting house, whose precise location the agency keeps secret for security reasons, 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines.

OPTION G

The precise location of the MTA's counting house, in which 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines, is kept secret by the agency for security reasons.

OPTION H

The agency keeps the precise location of the MTA's counting house, in which 29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines, secret for security reasons.

OPTION I

29 workers handle more than one million coins of all types deposited every day into bus fare boxes and rail ticket machines inside the MTA's counting house, whose precise location the agency keeps secret for security reasons.