The Goal of Language Understanding

Chapter 5: Dynamics of Language and Reasoning

John F. Sowa

VivoMind Research, LLC

1 March 2014

The Goal of Language Understanding

Outline:

1. Problems and challenges (goal.pdf)
2. Psycholinguistics and neuroscience (goal2.pdf)
3. Semantics of natural languages (goal3.pdf)
4. Wittgenstein's early and later philosophy (goal4.pdf)
5. Dynamics of language and reasoning
6. Analogy and case-based reasoning (goal 6.pdf)
7. Learning by reading (goal7.pdf)

Each chapter is in a separate file. Later chapters make occasional references to earlier chapters, but they can be read independently.

5. Dynamics of Language and Reasoning

Natural languages adapt to the ever-changing phenomena of the world, the progress in science, and the social interactions of life.

No computer system is as flexible as a human being in learning and responding to the dynamic aspects of language.

Three strategies for natural language processing (NLP):

1. **Neat: Define formal grammars with model-theoretic semantics that treat NL as a version of logic. Wittgenstein pioneered this strategy in his first book and became the sharpest critic of its limitations.**
2. **Scruffy: Use heuristics to implement practical applications. Schank was the strongest proponent of this approach in the 1970s and ’80s.**
3. **Mixed: Develop a framework that can use a mixture of neat and scruffy methods for specific applications.**

NLP requires a dynamic foundation that can efficiently relate and integrate a wide range of neat, scruffy, and mixed methods.

Need a Flexible, Dynamic Semantics

The mapping from language to the world uses all the capabilities of human intelligence and experience.

The usual model-theoretic semantics for logic is too rigid:

- A finite set of symbols with fixed definitions.
- Two-valued denotations {T, F}.
- A formal algorithm for computing the denotations.

A fixed set of word senses can be useful for a specialized task.

But no fixed set of senses defined by a fixed ontology can support the flexibility of human language and reasoning.

More generally, no discrete set of ontological categories can adequately represent a continuously variable world.

How could dynamic methods extend, revise, and supplement the logic and ontology?

Language and Reasoning

A knowledge base with a fixed ontology is too inflexible to answer questions that had not been anticipated.

Example:

- **Is Reno east or west of San Diego?**

A typical answer derived by deduction:

- **Reno is in Nevada, and San Diego is in California.**
- **Nevada is east of California.**
- **Therefore, Reno is east of San Diego.**

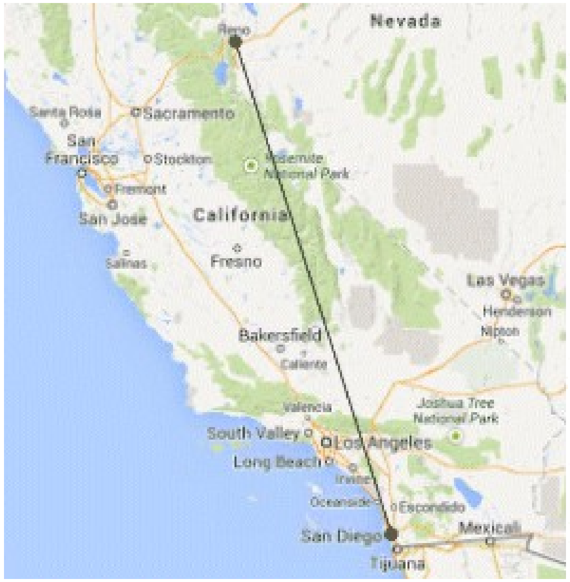
Another question:

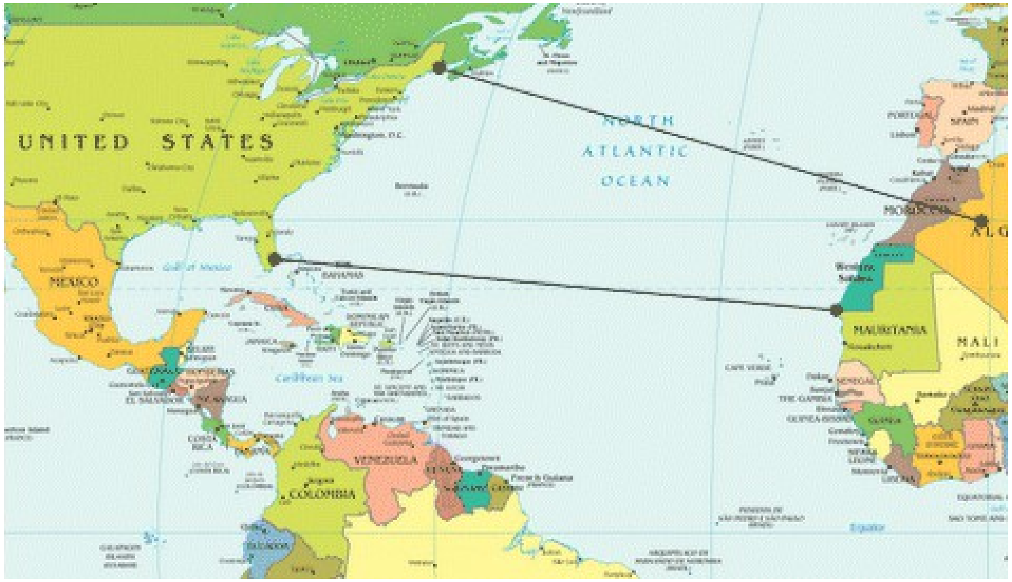
- **Which state of the US is closest to Africa?**

A typical answer:

- **Africa is southeast of the US.**
- **Florida is the southernmost state on the eastern seaboard.**
- **Therefore, Florida is the closest state to Africa.**

Reasoning With Images





Drawing lines on an image can make some aspects "obvious".

On a map of California and Nevada,

- **Draw a straight line from Reno to San Diego.**
- **Observe which city is farther west.**

On a map of the Atlantic Ocean,

- **Draw a line from Florida to Africa and keep the length fixed.**
- **Move the left endpoint to Maine, and rotate the line toward Africa.**
- **Observe where the right endpoint lies.**

Diagrammatic Reasoning

Some theorems are easier to prove than others.

Textbooks in mathematics usually make a 3-way distinction:

- Theorem: a major proposition that is difficult to prove.
- Lemma: a simpler, preliminary theorem that makes the major theorem easier to prove.
- Corollary: a theorem that is easy to prove after the major theorem has been proved.

Observation by C. S. Peirce:

- Every major theorem in Euclid has a new diagram.
- The proof of every corollary can be "seen" from the diagram.

The hardest task is the discovery of an appropriate diagram:

- Theorematic reasoning: an insight that leads to the diagram.
- Corollarial reasoning: immediate observations from the diagram.
- The fundamental insight (abduction) is embodied in the diagram.

Dynamic Semantics

To be computable, mental models, language games, and theories about them must be formally defined.

A Tarski style of model theory is too static to represent the dynamics of human thought, language, and reasoning.

The flexibility of language games and mental models requires a framework that can grow and change dynamically.

Issues in dynamic semantics:

- **Relating sentences in a text or discourse to one another, to the immediate context, and to background knowledge.**
- **Constructing a computable model of the semantic content that can evolve during a narrative or debate.**
- **Dynamic methods for transforming the models to reflect changes in the world and in language, thoughts, and plans about the world.**
- **Evaluating the truth or relevance of the models in terms of perception of a current situation or memories of past situations.**

Versions of Dynamic Semantics

Linguists and logicians developed theories about the dynamic aspects of language in discourse and narratives:

- **The first was Peirce, who called the operations on existential graphs "a moving picture of the action of the mind in thought" (Pietarinen 2005).**
- **Hintikka (1973) defined a *surface model* of a sentence S as "a mental anticipation of what can happen in one's step-by-step investigation of a world in which S is true."**
- **Kamp (1981) developed *Discourse Representation Theory* as a dynamic method of modifying a logical form to accommodate new information.**
- **Groenendijk and Stokhof (1991) developed *dynamic predicate logic* with theories and models that can be updated during discourse.**
- **Van Eijck and Visser (2010) surveyed other variants. ***

But the dynamic methods must be integrated with all forms of perception, thinking, learning, reasoning, talking, and acting.

*** See http://plato.stanford.edu/entries/dynamic-semantics/**

Discourse Representation Theory

DRT is a widely used version of dynamic semantics. *

- **Based on a logical form called Discourse Representation Structure (DRS).**
- **Has rules for building DRSes from a stream of sentences in discourse.**
- **Covers a wide variety of linguistic and logical issues including anaphora, generalized quantifiers, plurals, tense, and aspect.**
- **Defines a model-theoretic semantics for the DRS logic.**

Relating DRT to a broader range of cognitive science:

- **The structure of DRS is isomorphic to Peirce's existential graphs (EGs).**
- **Peirce and Kamp independently invented their notations while they were searching for a logical form that had a direct mapping to language.**
- **Kamp's rules for mapping language to DRS can also be adapted to EGs.**
- **The psychologist Johnson-Laird noted that Peirce's EGs and rules of inference are a good candidate for a neural theory of reasoning. ****

*** Hans Kamp & Uwe Reyle (1993) *From Discourse to Logic*, Dordrecht: Kluwer.**

**** Philip N. Johnson-Laird (2002) Peirce, logic diagrams, and the elementary processes of reasoning, *Thinking and Reasoning* 8:2, 69-95. http://mentalmodels.princeton.edu/papers/2002peirce.pdf**

Issues in Mapping Language to Logic

Hans Kamp observed that the features of predicate calculus do not have a direct mapping to and from natural languages.

Pronouns can cross sentence boundaries, but variables cannot.

- **Example:** ***Pedro is a farmer. He owns a donkey.***
- **$(\exists x)(\text{Pedro}(x) \wedge \text{farmer}(x))$. $(\exists y)(\exists z)(\text{owns}(y,z) \wedge \text{donkey}(z))$.**
- **There is no operator that can relate x and y in different formulas.**

The rules for scope of quantifiers are different.

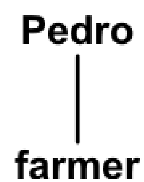
- **Example:** ***If a farmer owns a donkey, then he beats it.***
- **In English, quantifiers in the if-clause govern the then-clause.**
- **But in predicate calculus, the quantifiers must be moved to the front.**
- **Formula: $(\forall x)(\forall y)((\text{farmer}(x) \wedge \text{donkey}(y) \wedge \text{owns}(x,y)) \supset \text{beats}(x,y))$.**

In narratives, the default operator between NL sentences is usually equivalent to '*and then*'.

Linking Existential Quantifiers

Kamp invented Discourse Representation Structure (DRS) as a logic with a simpler mapping to and from NLs.

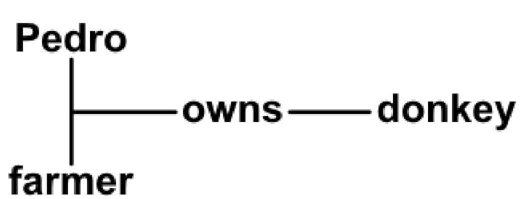
EGs (left) and DRSes (right) support equivalent operations.




```
x
Pedro(x)  farmer(x)
```


```
y   z
owns(y,z)  donkey(z)
```

By connecting EG lines or merging DRS boxes,



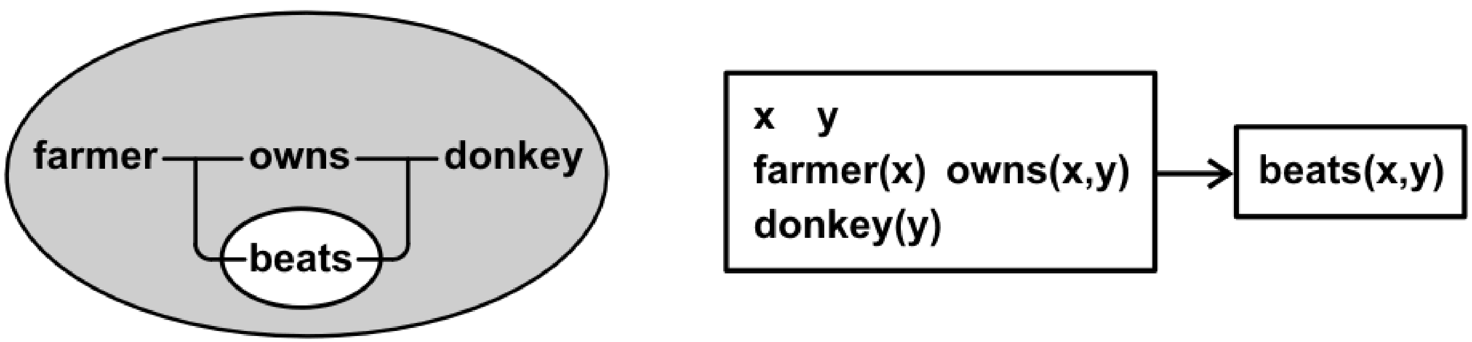

```
x   y   z   x=y
Pedro(x)  farmer(x)
owns(y,z)  donkey(z)
```

Quantifiers in EG and DRS

Peirce and Kamp independently chose isomorphic structures.

- **Peirce chose nested ovals for EG with lines to show coreference.**
- **Kamp chose boxes for DRS with variables to show coreference.**
- **But the boxes and ovals are isomorphic: they have the same constraints on the scope of quantifiers, and they have the same mapping to NLs.**

Example: ***If a farmer owns a donkey, then he beats it.***



In both EG and DRS, quantifiers in the *if*-area are existential, and they include the *then*-area within their scope.

Peirce's Rules of Inference

Three pairs of rules, which insert or erase a graph or subgraph:

- **Insert/Erase: Insert any graph in a positive (unshaded) area; erase any graph in a negative (shaded) area.**
- **Iterate/Deiterate: Iterate (copy) any graph in the same area or any nested area; deiterate (erase) any graph that could have been iterated.**
- **Double negation: Insert or erase a double negation (pair of ovals with nothing between them) around any graph in any area.**

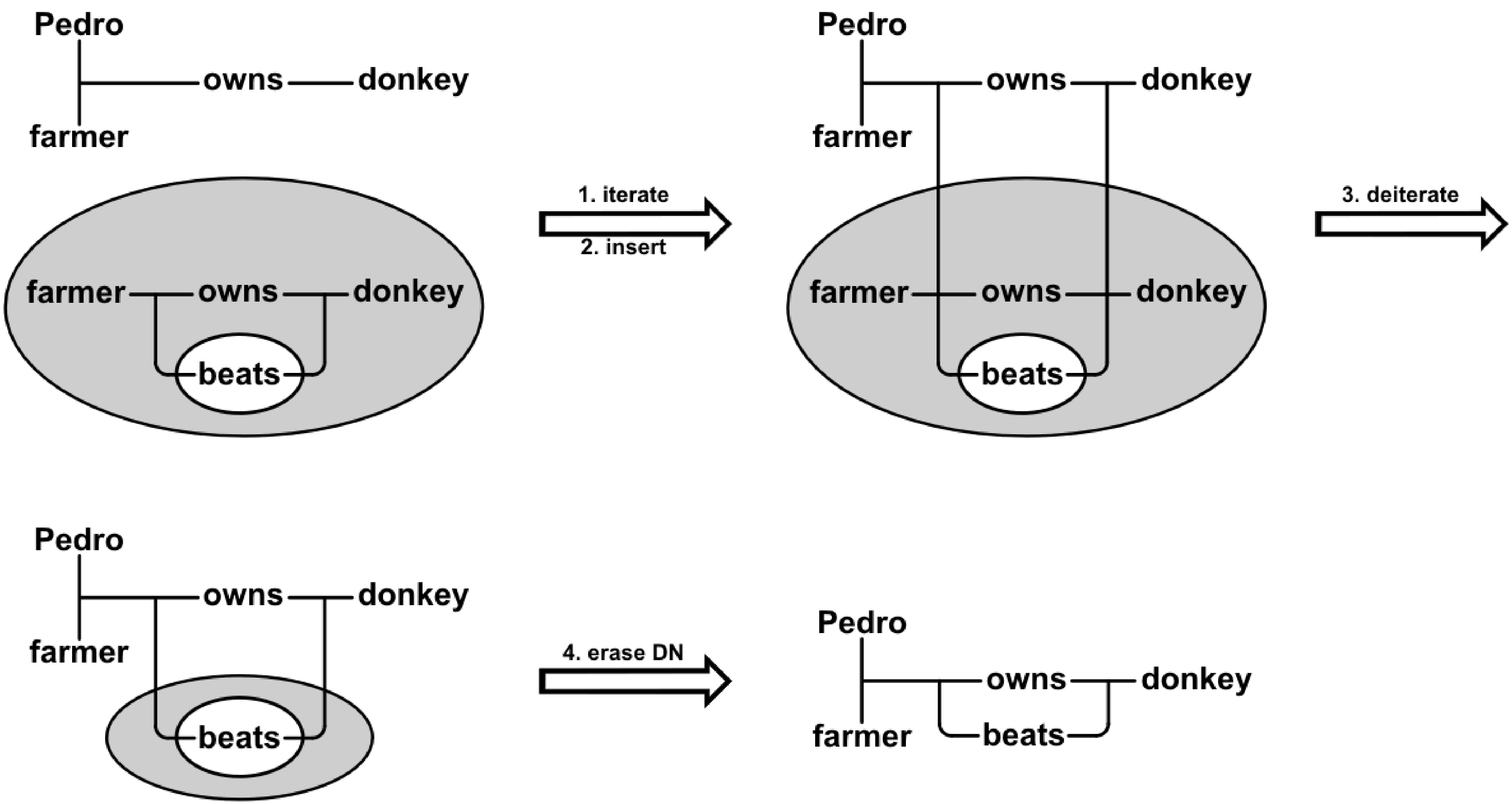
A four-step proof in the next slide:

1. **Iterate (extend) a pair of lines into a nested negative area.**
2. **Insert connections from those lines to lines in that negative area.**
3. **Deiterate (erase) a graph that is identical to one in an outer area.**
4. **Erase a double negation.**

Note that every step is a simple operation that could be performed by making or inhibiting connections between neurons. (See Lamb 2011.)

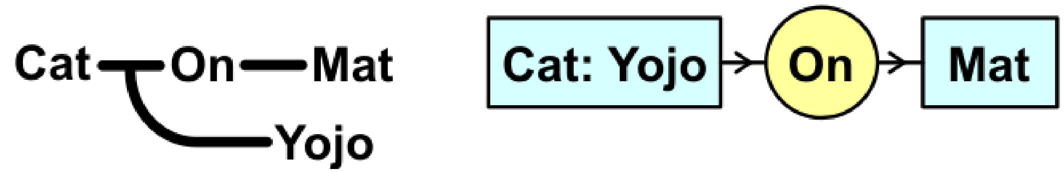
See Peirce's tutorial on EGs: http://www.jfsowa.com/pubs/egtut.pdf

A Proof by Peirce's Rules



Conclusion: ***Pedro is a farmer who owns and beats a donkey*.**

From EG to CG



Conceptual graphs add more features to the EG foundation:

- More symbols for Boolean operators and quantifiers.
- Point of quantification expressed by a concept box instead of a line.
- Concept box has a type field and a referent field, as in [Cat: Yojo].
- Types are represented by monadic relations or lambda expressions.
- Referent field may have names, indexicals, or generalized quantifiers.

A default ontology designed for mapping to and from NLs:

- Thematic roles and other relations used in linguistics.
- Context boxes for propositions and situations.
- Temporal, modal, causal, and intentional relations.

Operations adapted from DRS and other linguistic systems.

Mapping a Text to CG

A very short story:

> ***At 10:17 UTC, Yojo the cat and a mouse were in the basement of a house. Yojo chased the mouse. He caught the mouse. He ate the head of the mouse.***

Context-free translation:

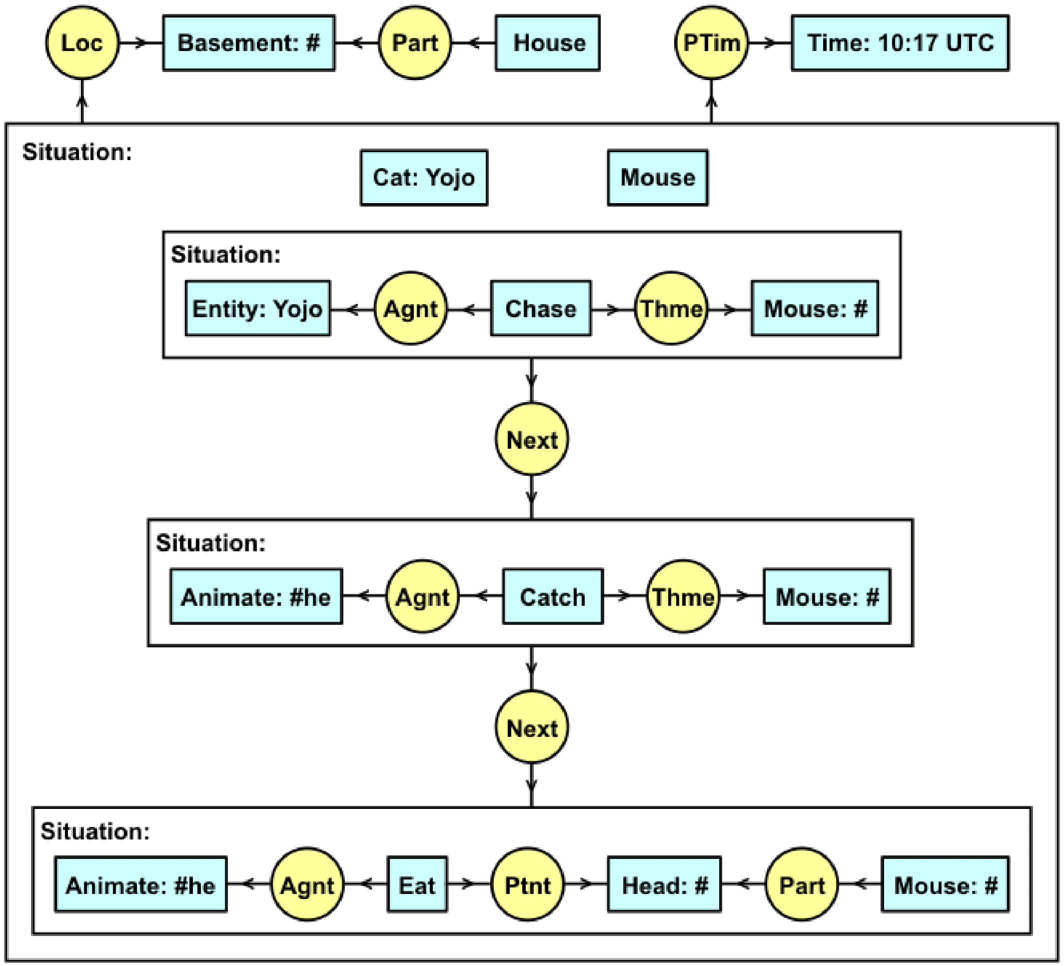
- **Translate each phrase of each sentence without considering the context.**
- **Leave some types and relations underspecified.**
- **Mark unresolved indexicals with the symbol #.**

As more information becomes available from the context,

- **Replace indexical markers with coreference labels.**
- **Specialize (or correct) the type labels on concepts and relations.**

Since DRS contexts are isomorphic to the EG and CG contexts, the DRS rules for indexicals can be adapted to CGs.

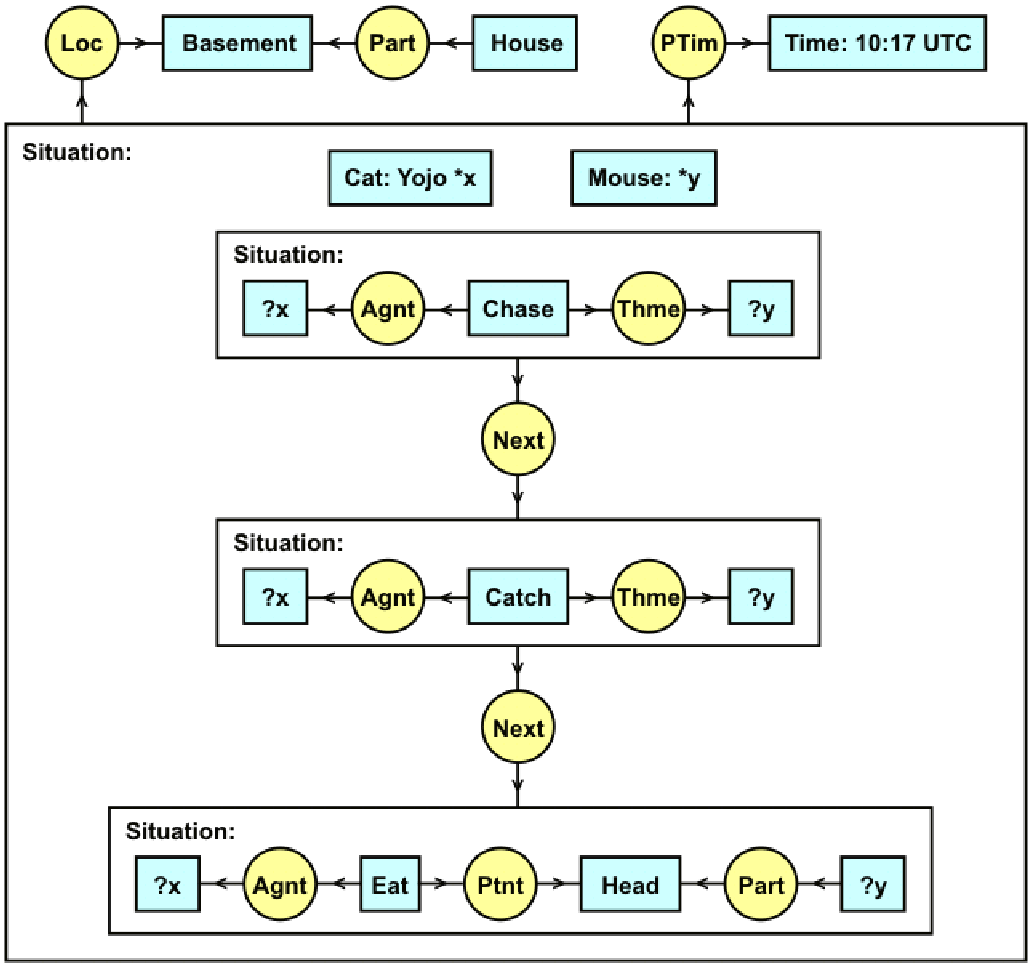
CG with Indexical Markers



The symbol # marks a concept with an unresolved indexical.

Each # marker must be replaced with a coreference label.

CG with Indexicals Resolved



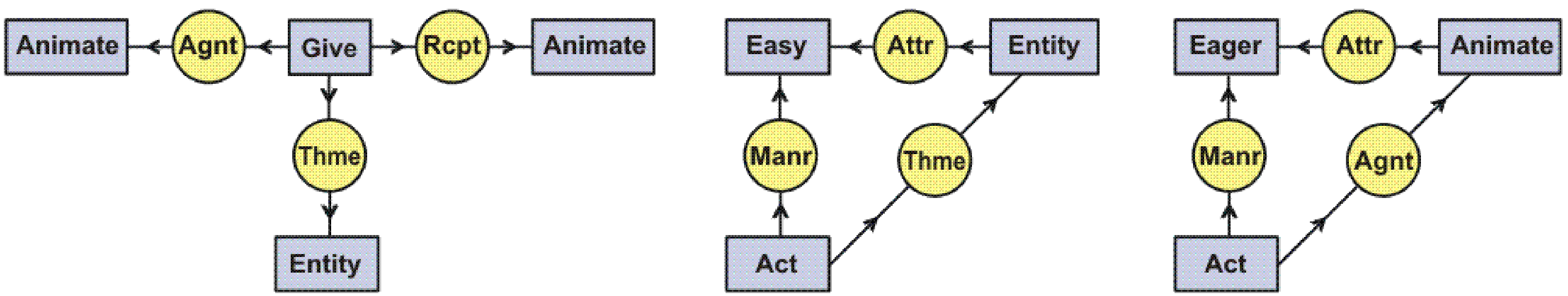
Symbols *x and *y are defining labels; ?x and ?y are bound labels.

In the pure graph notation, coreference is shown by lines.

Conceptual Schemata

A *conceptual schema* is a conceptual graph that represents a pattern that is typical for a given concept type.

Schemata for the concept types Give, Easy, and Eager.

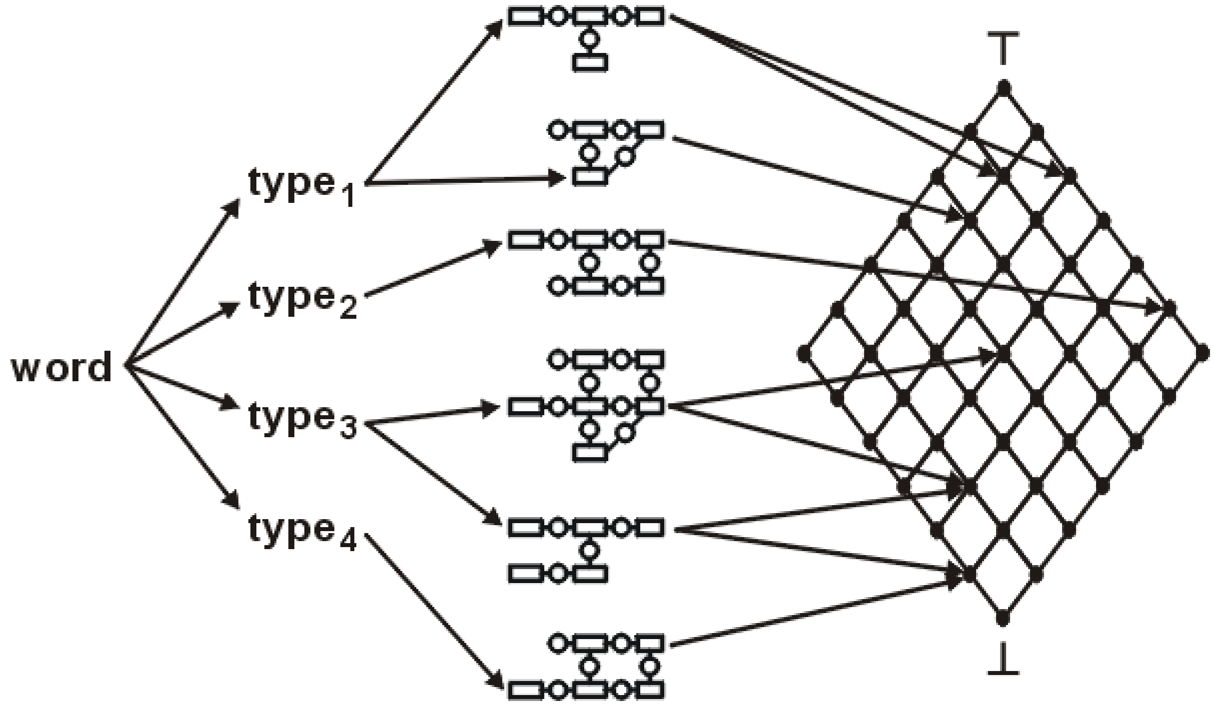


These graphs encode the typical patterns associated with each concept or relation type.

They can be used to select theories or other knowledge relevant to subjects described by those concepts.

The schemata for verbs specify the *case relations* or *thematic roles* and the constraints on concept types. See the IBM-CSLI verb ontology, http://lingo.stanford.edu/vso/

Mapping Word Fans to a Lattice of Theories



words → concept types → schemata → theories

The Continuum from Scruffy to Neat

DRT and other dynamic logics are neat and formally defined.

They can support useful versions of controlled NLs for narrowly defined language games. *

But nearly all unrestricted speech or writing contains unsolved research problems.

Yet many "scruffy" programs can extract useful information from such language, even though they don't have a formal semantics.

As Einstein and Halmos observed, even mathematicians start with informal language, which they gradually refine.

Any NLP system for unrestricted language must support an open-ended continuum from scruffy to neat language forms.

* ACE is a system for controlled English that uses DRS: http://attempto.ifi.uzh.ch/site/

Lattice of Theories

To support the full range of human language, a framework for NLP must be able to represent anything that anyone might ever say.

That framework must include all neat theories, and it must relate them to any kind of language, neat or scruffy.

For any version of logic L, theory X is more general than theory Y, and Y is more specialized than X, if and only if

- **X is true of everything (every model for L) for which Y is true.**

Generalization determines a *Lindenbaum Lattice* of theories:

- **If theory X is more general than Y, write $X \geq Y$ or $Y \leq X$.**
- **For any X and Y, there is a unique *minimal common generalization*, written $X \cup Y$, such that $X \cup Y \geq X$ and $X \cup Y \geq Y$.**
- **For any X and Y, there is a unique *maximal common specialization*, written $X \cap Y$, such that $X \cap Y \leq X$ and $X \cap Y \leq Y$.**
- **The most general theory at the top of the lattice, written $\top$, is true of everything (every model).**
- **The most specialized theory at the bottom, written $\bot$, is true of nothing.**

Model Theory for Mental Models

Every theory in the lattice except the bottom ⊥ has one or more Tarski-style models for which the theory is true:

- **Every model *m* has a set of entities D called the *domain* of *m*.**
- **D has a subset R of *relations* and a subset F of *functions*.**
- **An *atom* is a relation in R or a function in F with an entity in D assigned to each argument position (including the result position of a function).**
- **Some set of atoms is declared to be true.**

Mental models are images of continuous aspects of the world.

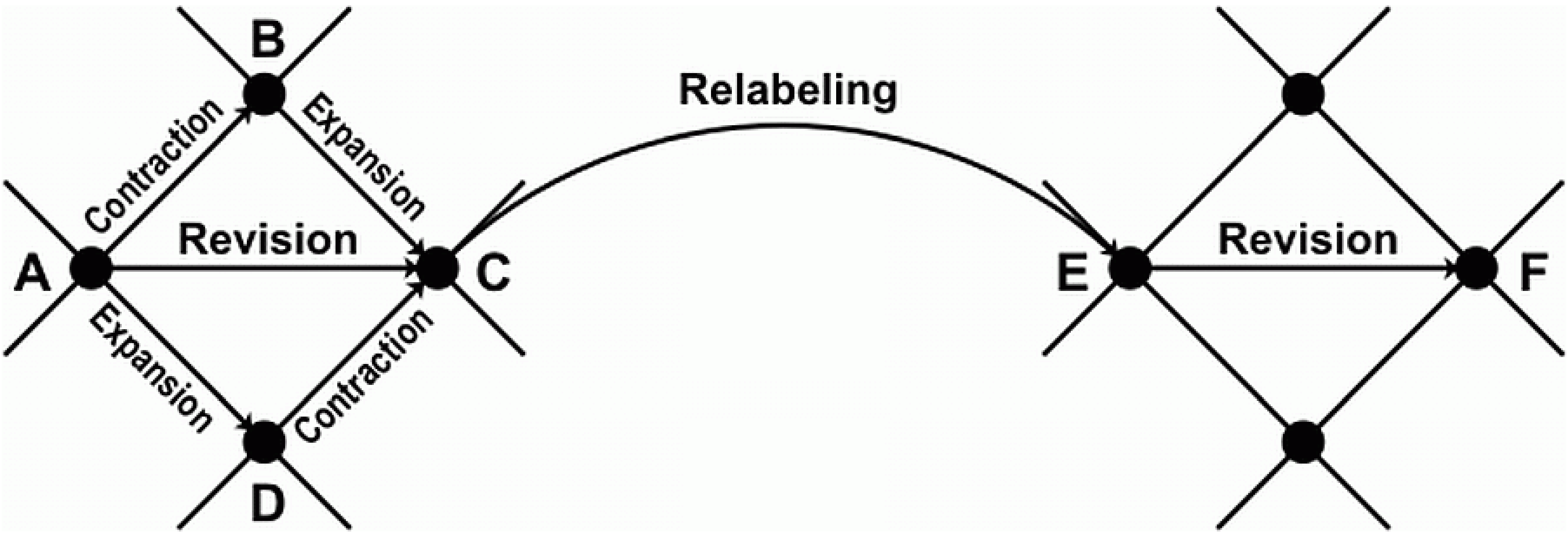
They can be described by discrete words or atoms.

But a complete description would specify which atoms are false:

- **Closed world assumption (CWA): Any atom not declared true is false.**
- **Open world assumption (OWA): Some atoms are declared true, some are declared false, and the truth values of the others are unknown.**
- **Dynamic semantics and nonmonotonic reasoning provide systematic methods for filling the gaps in the truth values of OWA models.**

Navigating the Lattice of Theories

Methods of belief revision for relating one theory to another:



Four operators: contraction, expansion, revision, and relabeling.

Every method of learning or nonmontonic reasoning determines a strategy for walking or jumping through the lattice of theories.

Learning and Belief Revision

Children learn language by starting with words and patterns of words that are linked to perception and action.

By trial and error, children and adults revise, extend, and adjust their beliefs to make better predictions about the world:

- **Observations generate low-level facts (atoms).**
- **Induction derives general axioms from multiple facts.**
- **A mixture of facts and axioms is an unstructured *knowledge soup*.**
- **Abduction selects some axioms to form a hypothesis (theory).**
- **Analogies relabel a theory of one subject and apply it to another.**
- **Deduction from a theory generates predictions about the world.**
- **Action tests a prediction against reality.**
- **The effects of the action lead to new observations.**

These steps correspond to a walk through a lattice of theories.

Learning a New Theory

Observations generate facts:

```
Tweety is a bird.    Tweety flies.
Daffy is a bird.     Daffy flies.
Hooty is a bird.     Hooty flies.
```

Induction derives general axioms from multiple facts:

```
Every bird flies.
Every flying thing is a bird.
For every x, x is a bird if and only if x flies.
```

Any one of these axioms can be added to a subset of the facts to generate the other facts.

Heuristics give a slight preference for "`Every bird flies.`"

But the other axioms cannot be ruled out.

New Information Triggers Belief Revision

New observation:

```
Vampy is not a bird. Vampy flies.
```

This observation rules out two options, leaving just one:

```
Every bird flies.
```

Another observation:

```
Tux is a penguin. Tux is a bird. Tux does not fly.
```

This observation restricts the universal quantifier:

```
Every bird that is not a penguin flies.
```

Learning and belief revision can be interpreted as walks through the lattice to find a more appropriate theory.

Proofs in Nonmonotonic Logic

A proof by any method of nonmonotonic logic can be mapped to a walk through a lattice of purely classical FOL theories.

Default logic by Reiter (1980) is a widely used example:

- **Each default theory has two kinds of axioms: classical and default.**
- **Any proof is a sequence of steps: S_0, S_1, …, S_n.**
- **Some steps are classical, and some use a default axiom.**
- **The collection of classical axioms defines some theory C in the lattice.**
- **Each default step S_i makes some assumption A_i in classical FOL.**
- **Adding A_i to the current classical axioms is a revision by expansion.**
- **When the proof ends at step S_n, the classical theory C has been expanded to a more specialized classical theory C′.**
- **In theory C′, the same conclusion can be derived by classical FOL rules.**

Negation as failure (NAF) is a variant of default logic in which any proposition that cannot be proved is assumed to be false.

Dynamic Model Theory

For any first-order logic L, there are two lattices:

- $\mathscr{L}$ is a Lindenbaum lattice of all theories expressible in L.
- $\mathscr{M}$ is the sublattice of theories in $\mathscr{L}$ that consist of conjunctions of atoms and negations of atoms.
- Every model m for any theory t in $\mathscr{L}$ is isomorphic to (in a one-to-one correspondence with) some theory in $\mathscr{M}$.

The lattice $\mathscr{M}$ can describe mental models. Walks through that lattice can describe the ways that models can evolve over time:

- Every theory t other than $\bot$ can state constraints that are true of every model in at least one sublattice of $\mathscr{M}$.
- A *narrative* is a time-ordered sequence of models along some walk through the models in a sublattice determined by some theory t.
- Debates among multiple speakers may cause walks or jumps through the lattice $\mathscr{L}$ that can cause irregular jumps among models in $\mathscr{M}$.

Modal Logics

For the semantics of modal logics, Kripke's possible worlds can be mapped to the lattice of theories and the lattice of models.

Dunn (1973) showed how with his semantics of laws and facts: *

- **For every Kripke world *w*, there are two theories called the *laws* of *w* and the *facts* of *w*.**
- **The laws of *w* are all propositions necessarily true of *w*.**
- **The facts of *w* are all true propositions about *w* – necessary or contingent.**
- **All the laws of *w* form a theory that is a generalization of all facts of *w*.**
- **The atoms of any model of *w* form a description of the world *w*.**
- **For any worlds w_1 and w_2, the world w_2 is *accessible* from w_1 if and only if the theory of laws of w_1 is a generalization of the theory of facts of w_2.**

The laws of any world can be partitioned into separate theories for different modalities, such as alethic, deontic, or epistemic.

*** See http://www.jfsowa.com/pubs/laws.htm and http://www.jfsowa.com/worlds.pdf**

Relating Theories to Mental Models

Mental models are more complex than abstract theories:

- They are intimately connected to perception, action, and feelings.
- Language that expresses them reflects all those nuances.
- Wittgenstein tried to capture “a form of life” with all its complexity.

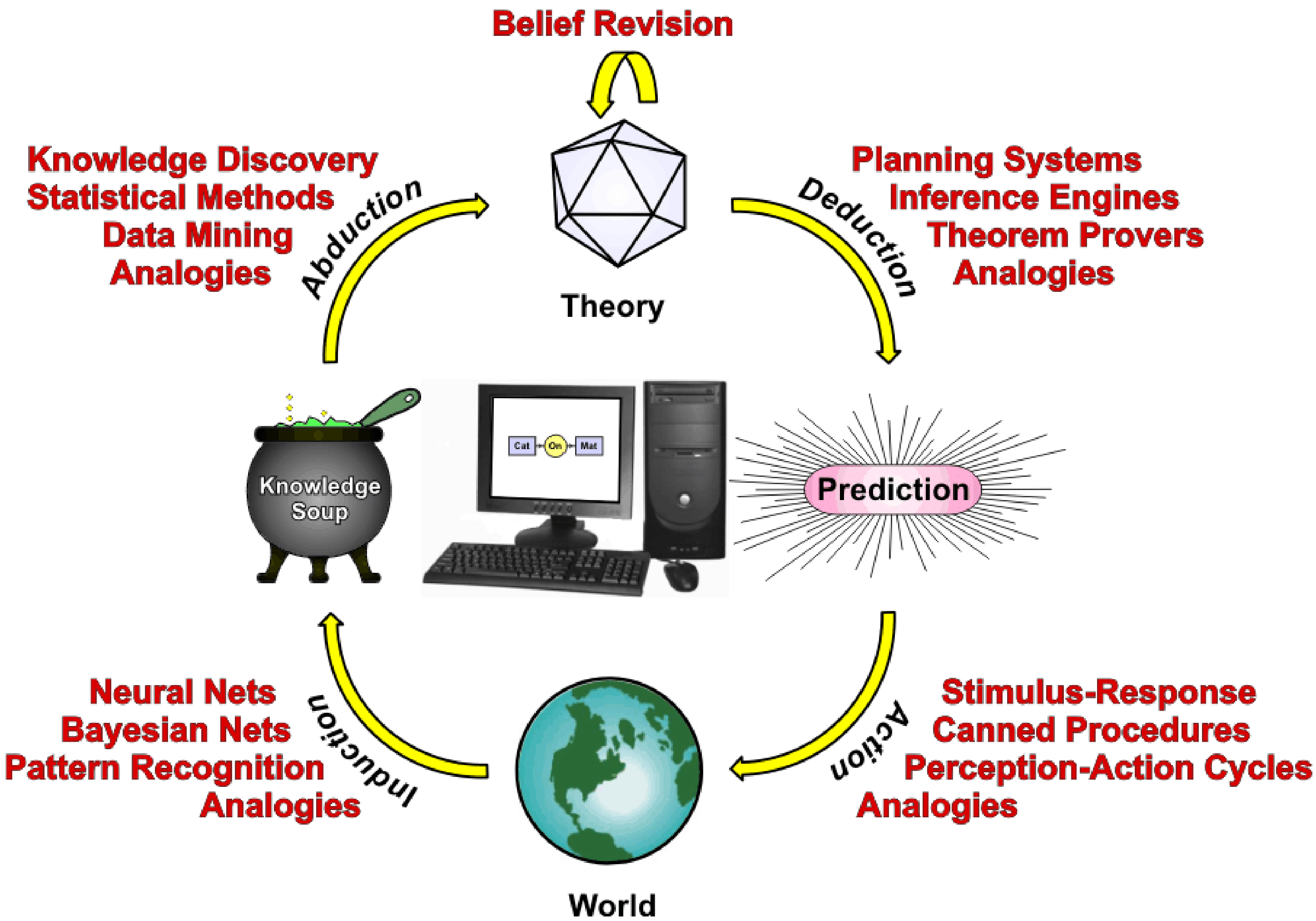
But the lattices are a better approximation than Tarski’s models:

- Instead of a fixed theory and model, the lattices support dynamic and systematic walks to find new theories and models as required.
- Dynamic changes in mental models ⇒ walks among models in $\mathcal{M}$.
- Learning and nonmonotonic reasoning ⇒ walks among theories in $\mathcal{L}$.
- Various computational methods can be used to determine the walks.

Only finite subsets of the lattices can ever be implemented, but the framework allows new theories and models to be computed:

- Conflicts, contrasts, and disagreements among multiple points of view (theories) can be detected, debated, resolved, or abandoned.

Peirce’s Cycle of Pragmatism



This diagram, based on Peirce’s writings, shows tools that can support neat and scruffy methods for walking through a lattice of theories.

See the OODA loop and the Albus cycle in slides 44 and 45 of goal2.pdf.

Processing the Knowledge Soup

Theories in the lattice are abstractions from the more flexible and vastly more complex connections in the brain.

Structured theories are crystallized from an unstructured soup processed by many different neural mechanisms.

The society of mind by Minsky (1986, 2006):

- There can be no single, unified theory of all neural computation.
- The brain contains many areas specialized for different purposes.
- Learning creates more specializations for every human need.
- The result is a society of interacting heterogeneous modules.
- Knowledge consists of a network of K-lines that connect related aspects in areas of the brain with different specializations.
- Emotions are the driving forces that motivate all operations.

Peirce’s cycle can accommodate a heterogeneous mixture of formal and informal computational modules and methods.

Peirce, Minsky, and Wittgenstein

Peirce's cycle relates the many ways of traversing the lattices:

- Induction, abduction, and deduction are special cases of analogy.
- The cycle may have nested cycles at different levels of granularity.
- Milliseconds: Observe danger, orient, decide, and act (OODA loop).
- Seconds to minutes: Routine analysis, planning, and problem solving.
- Days to years: Complex studies, exploration, and research.

Minsky's modules can process anything by any method:

- Heterogeneous representations in different areas of the brain.
- Multiple formal and informal computational methods.
- Asynchronous message passing for control and communication.

Wittgenstein's language games support open-ended flexibility:

- Dynamic methods relate the games to perception, action, and reasoning.
- They can be supported by a dynamic model theory.

Relating Language to the World

Language is directly related to perception and mental imagery.

- The only "language of thought" is imagery, including auditory images such as "inner speech" or an imagined melody.
- Peirce's graphs, Lamb's networks, and Minsky's K-lines are related hypotheses about the links that constitute knowledge in the brain.
- Language understanding uses perceptual mechanisms to interpret speech or writing in terms of those networks.

Semantics is grounded in mental models of the world.

- The continuous world is mapped to continuous mental images.
- As Peirce said, "symbols grow." Word meanings adapt to the continuity by an open-ended variety of microsenses.

Pragmatics is integrated with all social interactions.

- The great apes have a complex social life without a human language.
- But language facilitates social interactions – even for apes that learn a rudimentary subset. (Greenspan & Shanker 2004).

What is Language Understanding?

Understanding a text in some language does not require a translation to a language of thought or logical form.

Instead, it requires an interpreter, human or robot, to relate the text to his, her, or its context, knowledge, and goals:

- That process changes the interpreter's background knowledge.
- But the kind of change depends critically on the context, goals, and available knowledge.
- No two interpreters understand a text in exactly the same way.
- With different contexts, goals, or knowledge, the same interpreter may understand a text in different ways.

The evidence of understanding is an appropriate response to a text by an interpreter in a given situation.

If a robot responds appropriately to a command, does it understand? What if it explains how and why it responded?

Related Readings

Future directions for semantic systems,
http://www.jfsowa.com/pubs/futures.pdf

From existential graphs to conceptual graphs,
http://www.jfsowa.com/pubs/eg2cg.pdf

Role of Logic and Ontology in Language and Reasoning,
http://www.jfsowa.com/pubs/rolelog.pdf

Fads and Fallacies About Logic,
http://www.jfsowa.com/pubs/fflogic.pdf

Conceptual Graphs for Representing Conceptual Structures,
http://www.jfsowa.com/pubs/cg4cs.pdf

Peirce's tutorial on existential graphs,
http://www.jfsowa.com/pubs/egtut.pdf

ISO/IEC standard 24707 for Common Logic,
http://standards.iso.org/ittf/PubliclyAvailableStandards/c039175_ISO_IEC_24707_2007(E).zip

References

For more information about the VivoMind software:

Majumdar, Arun K., John F. Sowa, & John Stewart (2008) Pursuing the goal of language understanding, http://www.jfsowa.com/pubs/pursuing.pdf

Majumdar, Arun K., & John F. Sowa (2009) Two paradigms are better than one and multiple paradigms are even better, http://www.jfsowa.com/pubs/paradigm.pdf

Sowa, John F. (2002) Architectures for intelligent systems, http://www.jfsowa.com/pubs/arch.htm

Sowa, John F., & Arun K. Majumdar (2003) Analogical reasoning, http://www.jfsowa.com/pubs/analog.htm

Sowa, John F. (2003) Laws, facts, and contexts, http://www.jfsowa.com/pubs/laws.htm

Sowa, John F. (2005) The challenge of knowledge soup, http://www.jfsowa.com/pubs/challenge.pdf

Sowa, John F. (2006) Worlds, models, and descriptions, http://www.jfsowa.com/pubs/worlds.pdf

Sowa, John F. (2011) Cognitive architectures for conceptual structures, http://www.jfsowa.com/pubs/ca4cs.pdf

Related references:

Johnson-Laird, Philip N. (2002) Peirce, logic diagrams, and the elementary processes of reasoning, *Thinking and Reasoning* 8:2, 69-95. http://mentalmodels.princeton.edu/papers/2002peirce.pdf

Lamb, Sydney M. (2011) Neurolinguistics, Class Notes for Linguistics 411, Rice University. http://www.owlnet.rice.edu/~ling411

Harrison, Colin James (2000) *PureNet: A modeling program for neurocognitive linguistics*, http://scholarship.rice.edu/bitstream/handle/1911/19501/9969261.PDF

For other references, see the combined bibliography for this site:

http://www.jfsowa.com/bib.htm