Polysemy, Analogy, and Metaphor

James Foster

Prepared for the University Scholars Program, 2008-2009

Mentor Hana Filip, University of Florida

The main goal of this project is to explore the relationship between metaphors in language and analogical reasoning. We intend to provide answers to the following questions: How are word senses extended by analogy? What are the triggers of analogical comparison and inference? What is the role of abstraction in analogical domain mapping? We find that metaphor is a bridge between analogy and polysemy. The motivation behind these questions is to explain the use of polysemous words, such as the word 'tree' in computer science and biology, and to use these shared words to unify domains.

Introduction

In computer science the word ‘tree’ signifies an abstract data structure which has a root node, branches, and leaf nodes (Sahni 2005). But in fact the word ‘tree’ has other senses, such as the biological notion of a woody plant. A biological tree also has roots, branches, and leaves. In my introductory computer science (CS) class, the tree data structure was introduced alongside a picture of a biological tree. The comparison helps students understand the more abstract data structure in terms of more familiar, concrete woody plants.

Furthermore, CS trees also have the concepts of parent nodes and child nodes. Since tree nodes are certainly not birthing and rearing other tree nodes, why use the same terminology as biological reproduction? CS imports the words ‘parent’ and ‘child’ to name connected nodes in a hierarchal data structure, much like a human parent and child are connected in a hierarchal family genealogy. We might even say the connected nodes are ‘related’.

The CS and biological senses of a word such as tree are related by an analogy. When computer scientists needed a name for an acyclic connected graph,
rather than coining an entirely new word, they *extended* an existing word from another domain. A source domain (such as Biology) can provide a concrete understanding of a more abstract target domain (such as CS) (Gentner 2001; Sanders and Thagard 2004; Winston 1980; Lakoff and Johnson 1980).

Interestingly, the branching projections that a neuron uses to receive electrochemical input from other neurons are named *dendrites*, which is the Greek word for *tree*. This case, along with many other examples such as *soma* (“body”), *neocortex* (“new bark”), *nucleus* (“nut”), and *kernel* (“grain”) supports the argument that scientific phenomena are often named for the concrete objects they resemble.

In order to learn new uses of a polysemous word such as *tree*, we can make an analogy to more familiar senses of *tree*. The analogy acts as a bridge between the structures we want to learn and the structures we already understand. Framing a novel problem in terms of a well-understood, pre-stored problem can help us find a creative solution in unfamiliar territory (Dunbar 2001; Gentner 2001).

The study of metaphor and analogy has important applications in science, education, and language understanding.

Analogical thinking is a key component of all aspects of scientific reasoning, ranging from hypothesis generation to experimental design, data interpretation, and explanations. Analogy not only permeates all aspects of scientific thinking, but is a key component of the ways that scientists reason about unexpected findings. (Kevin Dunbar, The Analogical Paradox, pg. 3)

If you push the novelty of language and metaphor far enough, you can end up with a new way of seeing... which can in its own right make an original contribution to science... Vivid metaphors fueled Einstein’s creative genius. (Richard Dawkins, The Selfish Gene, pg. xvi)
In addition to analogy, the study of metaphor requires an examination of the categorizing/division of a word’s meanings into discrete senses. It is not clear how word senses should be enumerated for natural language processing applications such as word sense disambiguation, machine translation, information retrieval, question answering, text summarization, and language understanding (Palmer et al, 2005). Nor is it clear how people mentally structure word meanings (Brown 2008; Tuggy 1993; Geeraerts 1993).

**Main Questions**

If we consider that related senses of a word are activated by exposure to one particular sense (Brown 2008), then one sense of a word might act as a guide to interpreting new uses of that word. In technical domains such as computer science, many names are imported from other domains. These imported names are frequently metaphorical extensions of a previously established sense in a source domain. As a result, the name of an entity can provide a clue to its underlying structure. For example, the names of data structures (*tree, array*) hardware (*gate, hub*), and other CS names such as *semaphore, signature,* and *key,* can all be understood in terms of more common objects and experiences with the real world. This phenomenon of imported terminology can be harnessed to facilitate learning by providing a framework for understanding unfamiliar terms in a target domain.

We would like to examine how word senses are extended by analogy, how analogical comparison is triggered, and how the process of abstraction enables mapping between source and target domains.

In order to answer these questions, we must first review previous literature on word senses, metaphor, and analogy. First, we will see that a word’s senses typically fall on a continuum of relatedness and cannot be neatly categorized as either polysemous (related) or ambiguous (unrelated). Second, we will examine metaphoric change in meaning, and see how the novel metaphoric use of a
word can become lexicalized as another conventional sense. Finally, we will review abstraction in three components of analogical reasoning – access, mapping, and inference.

**Word Senses**

A critical challenge in linguistics is to determine what is a word sense. Many (if not all) words have multiple meanings/usages, which can generally be classified as related or unrelated meanings. As an example, consider the word *hub*, which can refer to (1) the central part of a bicycle wheel, (2) the central part of a motorcycle wheel, (3) a computer networking device which other devices connect to, and (4) the central location of a transportation or commerce system. Although we have divided *hub’s* senses into these four categories (which are not exhaustive), we could actually combine or divide these four senses further. The sense divisions depend on how specific or general each sense category is defined. Certainly, sense (1) and (2) are very closely related, since each refers to the central part of a wheel. It does not seem reasonable to enumerate a separate sense for the wheels of every kind of vehicle because these usages of *hub* are so similar. Instead, we should probably combine sense (1) and (2) into a single, slightly more abstract sense: the central part of a wheel.

Linguists use the word *polysemy* to refer to senses that are related, and *ambiguous* to refer to unrelated senses. Senses can be tested for relatedness by assessing the semantic oddness known as *zeugma* (Tuggy 1993) in a sentence with parallel structure such as “American Airlines serves peanuts and Atlanta”. In this case, we can tell that *serves* is ambiguous because of the resulting zeugma. Quantifying the degree of ambiguity is a challenge for both cognitive and computational linguistics.

The word *vagueness* is used to describe words such as ‘a lot’ and ‘many’, which can refer to different quantities depending on who utters them. One person’s
concept of “a lot of potatoes” might be 5 potatoes, but to a farmer “a lot of potatoes” might be 10 sacks.

WordNet is a lexical database for English commonly used for natural language processing applications. However, for some uses, WordNet’s sense divisions are too granular (Jurafsky 2007; McCarthy 2006). We would like to be able to group WordNet senses in a way that improves computer performance in word sense disambiguation, machine translation, information retrieval, and language understanding. The Unified Verb Index groups WordNet senses and links the groups to other linguistic databases - Propbank and FrameNet - to provide a hierarchy of linked sense divisions (Kipper et al. 2008). This hierarchy can provide for different levels of sense granularity depending on the needs of a specific application.

**Analogy**

In Structure Mapping Theory, Gentner decomposes the process of analogy into three sub-processes: *access, mapping, and inferential power* (Gentner 1985).

**Access**

Analogical access addresses the question of how a given situation reminds someone of a previous situation stored in his or her memory.

What determines which comparisons people will think of spontaneously?... We know very little about how people notice analogies. When faced with a target problem or situation, what are the factors that cause a person to spontaneously think of some piece of stored knowledge? The dominant position in artificial intelligence research access occurs via shared causal relations or other higher-order relations... *Access* is the process of matching a base situation in memory with a given target situation a person is faced with. In other words, it is the process of a given target situation reminding a person of a base situation in his memory (Gentner 1985).
Gentner found that “accessibility is governed by literal or surface similarity, not similarity of higher-order structures” and that “inferential power is governed by similarity of higher-order structures... analogical access and analogical inference are governed by very different rules. This calls into doubt the causal indexing view held by many researchers in artificial intelligence” (1985).

However, after much time spent observing scientists in their labs, Kevin Dunbar found that processing can depend on conventionality of analogy, previous exposure, storage method and retrieval cues:

Naturalistic environments make it possible for people to use structural features and higher order relations because information is encoded in a richer way. Furthermore, naturalistic settings influence the retrieval conditions, often stressing the search for higher-order analogs rather than analogs that share superficial features. Experiments on analogy have tended to use conditions that do not stress a rich encoding of the information, nor stress structural information at retrieval... When generating analogies people search memory for structural relations, but when they are asked to choose between different sources they will focus on superficial features (Dunbar 2001).

Dunbar’s method here is superior to Gentner’s because it examines spontaneous analogies rather than forcing a choice among given options.

**Mapping**

It is not clear how analogical domain mapping happens in human cognition. Three suggested models of directionality and symmetry in (Gentner 2001) are initial temporal asymmetry, initial processing asymmetry, and initial symmetry followed by processing asymmetry.
(1) Initial temporal asymmetry – processing begins with the base; after information is accessed or abstracted from the base, it is projected from the base to the target.

(2) Initial processing asymmetry – processing begins simultaneously with both terms, but is differentiated from the start in role-specific ways (Glucksberg, McGlone, and Manfredi 1997).

(3) Initial symmetry followed by processing asymmetry- the initial stage is a role-neutral alignment stage; it is followed by a directional process of inference projection.

Gentner provides evidence supporting initial symmetry followed by processing asymmetry with the reversed metaphor inference effect: metaphors took longer to reject than ordinary false statements. For example, *Some handcuffs are contracts* vs. *Some contracts are handcuffs*. But the key finding was that reversed metaphors showed just as much interference as forward metaphors (Wolff and Gentner 2000).

**Inferential Power**

Initial mapping(s) between known entities and relations of a source and a target domain are established can lead to spontaneous inference by analogy (Gentner 2001). Structures in the target domain can be inferred by carrying entities and relations from the source domain over to the target domain. It is important to note that source domain relations can be tweaked or modified in order to make sense in the target domain. Transformations can be made to the source domain structures to fit the target domain. Gentner (2001) argues that systematic and consistent structures are the structures more likely to be carried over to the target domain.

The implementation of Structure Mapping Theory is Structure Mapping Engine (Falkenhainer, Forbus, and Gentner 1989).
SME uses local-to-global alignment process to arrive at a structural alignment of two representations. First, SME begins blind and local by matching all identical predicates in the two representations. Semantic similarity between predicates is captured through a decomposition into partial identities. These are typically inconsistent, many-to-one matches. Second, local matches coalesced into structurally consistent clusters (kernels) that depend both on sheer number of predicates and depth of the system. Third, kernels are merged into one or a few structurally consistent global interpretations (mappings displaying one-to-one correspondences and parallel connectivity). SME uses a greedy merge algorithm (in linear time) to avoid exhaustive search.

Next, SME produces structural evaluation of the interpretation(s) that favors deep systems over shallow systems. Then (losing role-neutrality), predicates connected to the common structure in the base, but not initially present in the target, are projected as candidate inferences in the target. Thus, structural completion can lead to spontaneous unplanned inferences.

One criticism of the implementation of Structure Mapping Theory, the Structure Mapping Engine (Falkenhainer et al. 1986) is that the source and target domain are already given to the system. The process of finding an appropriate source domain to map to a target domain is ignored. In addition, the source and target domain representations are built into the system. In contrast, the analogy system CopyCat (Hofstadter 1995) constructs its micro-domain representations dynamically. Although CopyCat frequently behaves in a realistic human manner, it is purposefully not tested against human performance data.

\textit{Abstraction}
A good definition of abstraction is “the process of keeping what is ‘essential’ and discarding everything else”. Abstraction is a fundamental problem solving technique in computer science. Examples include abstraction by parameterization, abstraction by specification, data abstraction, procedural abstraction, and iteration abstraction (Liskov 2001).

Do we abstract from situations (or a single situation) first, and then recognize other concrete instances of the abstraction? Or, do we notice some similarity between concrete situations and then generalize to some abstraction that is general or flexible enough for both situations? What is the similarity that we notice? Is it shared terminology, other surface similarity, or deeper structural similarity? Abstraction may be mediated by metaphor. In the case of polysemous domains, the abstract structure may be what the polysemous senses have in common.

There are some modifications or abstractions that must be applied to, say, an oak tree, in order to compare it to the data structure tree. CS trees are typically drawn upside down, with the root at the top and the leaves at the bottom. We ignore many details of the oak tree, such as its bark, color, size, and the fact that its roots are underground. In the data structure, nodes are usually represented with a circle and links with a thin line.

Analogy involves parameterizing a situation by making particular, specific entities *variables* which can be filled by different *arguments* to the “situation function”. These parameter “slots” can be filled by different specific, concrete entities. We might say that all the different particular arguments that can fill a different slot share some abstraction. The abstraction represents what the specific arguments have in common that is relevant to the situation function.

In object-oriented programming, an abstract type or interface can be implemented by concrete classes. The abstract class contains the data and operations that the concrete classes have in common. With polymorphism, the
concrete class can be used through its abstract interface without knowledge of its specific implementation.

However, it is not clear that this sort of object-oriented abstraction process takes place in the brain. Hawkins (2004) describes a theory of the patterns of abstraction in the layers of the neocortex.

... A similar abstraction of form is occurring throughout the cortex, in every region. This is a general property of the neocortex. Memories are stored in a form that captures the essence of relationships, not the details of the moment. When you see, feel, or hear something, the cortex takes the detailed, highly specific input and converts it to an invariant form. It is this invariant form that is stored in memory, and it is the invariant form of each new input pattern that it gets compared to. Memory storage, memory recall, and memory recognition occur at the level of invariant forms. There is no equivalent concept in computers. (Hawkins 2004, pg. 82)

Metaphor

The Oxford Concise Dictionary of Linguistics defines one sense of metaphor as:

A figure of speech in which a word or expression normally used of one kind of object, action, etc. is extended to another. This may lead to metaphorical change in meaning: thus what is now the normal sense of lousy is in origin a metaphorical extension from the basic sense ‘full of lice’ (Matthews 2007, pg.243).

We are interested in this metaphorical change in meaning as the process behind word sense extension by analogy. A metaphor can be considered the verbal form of an analogy. Lakoff and Johnson (1980) detail the existence of systematic conceptual metaphors such as LOVE IS A JOURNEY and ARGUMENT IS WAR. Lakoff and Johnson claim that there is no abstraction that could underlie systems of metaphors such as LOVE IS A JOURNEY, LOVE IS WAR, LOVE IS A PHYSICAL FORCE, LOVE IS MADNESS. However, in order to make these comparisons, irrelevant details of each of the source and target
domains must be ignored. While there may be no single abstract concept that underlies each of the LOVE metaphors, there is an abstraction process that enables the comparison.

Our proposal is that words can act as a guide or constraint to finding a source domain. Unfamiliar or novel senses of a word from a target domain can act as a clue for examining other, preexisting senses of the same word in potential source domains. This is a significant constraint, but in many cases, other senses make good source concepts because many concepts are named by metaphorical extension of other senses.

**Experiments**

It has been shown that exposure to one sense of a word primes a person’s response time to another sense in proportion to how related the two senses are (Brown 2008). We would like to test the manner in which polysemous senses are related. One of the main thesis of this project is that polysemous senses have an abstract analogical relationship.

One possible experiment is to randomly assign subjects to two conditions, and give each condition two tests. In the first test, present first condition subjects with a picture or movie that depicts some novel phenomenon, and have the subjects describe what they see verbally. Our hypothesis is that subjects would import terminology from other domains to describe the unfamiliar relationships and events in the picture or movie.

In a second condition, subjects would view the same picture or movie, and rather than describe the events verbally, they would solve a problem about it that requires deep structural understanding without requiring a verbal description.

In a second test, subjects in the first condition would solve a deep structure problem in a domain that shares terminology with the words the subjects used
to describe the picture or movie in the first test. If the words they originally used have senses that are related by an abstract analogy, the subjects should be able to solve the problem faster and more accurately than a control group who has not been exposed to the descriptions of the picture or movie.

Subjects in the second condition would complete a lexical decision task in which they have to decide if a sequence of characters is a real word or not. They would respond to key words that the first condition subjects used to describe the picture or movie with carefully selected control words interlaced into the key word sequence. If solving a deep structure problem about the picture or movie primes the words that describe it, even if the subjects did not use those words while solving the problem, the subjects should respond faster and more accurately to the key words than the control words.

Another test would be to randomly assign subjects to conditions A and B. In condition A, subjects solve a complex problem without describing the problem verbally or communicating their solution. In condition B, subjects would solve the same complex problem, but would then describe the problem and their solution verbally to the experimenter. In a second phase, both conditions would solve another problem that is analogically related to the first problem. My hypothesis is that the subjects who used words to describe the problem and communicate their solution would be better at solving the analogically related problem, because the words they used to describe the first problem would guide them in solving the second problem.

**Implementation**

A good implementation of metaphor or analogy needs an efficient mechanism of analogical access. In the Metaphor Extension System (MES), Martin addresses the access issue by having an explicit knowledge base of conventional metaphors (Martin 1990). When a novel metaphor is encountered, the concept hierarchy is used to constrain candidate metaphors. The situation is
somewhat different than typical analogical access because MES is extending entire metaphor names, and searching for preexisting metaphors as candidate metaphors for extension. MES deals with metaphors at the verb level in the Unix Consultant domain, such as *kill a process*. It would be interesting to develop a similar system for nouns, which maps structures of a noun such as *tree* to the structures of other senses of the same word.

**Conclusion**

We have examined several examples of words with analogically related senses, and seen that abstract concepts can be understood in terms of more concrete experiences. This basic thesis of word sense extension by analogy requires the study of word senses, metaphoric change in meaning, and analogical reasoning. It is clear that the process of abstraction plays an important role in analogy and metaphor because comparison requires irrelevant details to be ignored.

**References**


Gamma, Helm, Johnson, and Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. (1995)


Jurafsky and Martin. *Speech and Language Processing*. (2009)


Lakoff and Johnson. *Metaphors We Live By*. (1980)


Palmer, Dang, and Fellbaum. “Making fine-grained and coarse-grained sense distinctions, both manually and automatically”. *Natural Language Engineering*. (2005)


