How to make the Semantic Web more semantic

Peter GÄRDENFORS
Lund University Cognitive Science
Kungshuset
S-222 22 Lund, Sweden
Peter.Gardenfors@lucs.lu.se

Abstract. The Semantic Web is not semantic. It is good for syllogistic reasoning, but there is much more to semantics than syllogisms. I argue that the current Semantic Web is too dependent on symbolic representations of information structures, which limits its representational capacity. As a remedy, I propose *conceptual spaces* as a tool for expressing more of the semantics. Conceptual spaces are built up from quality dimensions that have geometric or topological structures. With the aid of the dimensions, similarities between objects can easily be represented and it is argued that similarity is a central aspect of semantic content. By sorting the dimensions into domains, I define properties and concepts and show how prototype effects of concepts can be treated with the aid of conceptual spaces. I present an outline of how one can reconstruct most of the taxonomies and other meta-data that are explicitly coded in the current Semantic Web and argue that inference engines on the symbolic level will become largely superfluous. As an example of the semantic power of conceptual spaces, I show how concept combinations can be analysed in a much richer and more accurate way than in the classical logical approach.

1 The dream of the Semantic Web

Given the enormous amount of information on the Internet, it is becoming ever more important to find methods for information integration. The Semantic Web is the most well known recent attempt in this direction. In an introductory article, Berners-Lee, Hendler and Lassila [3] write that "the Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." The ambition of the Semantic Web is excellent, but, so far, most work has been devoted to developing languages such as RDF for representing information and OWL for expressing ontologies. In my opinion, to enable "computers and people to work in cooperation" one should, above all, take into consideration how *humans* process concepts. As I shall argue in this article, this will be necessary if we want to put real semantic content into the Semantic Web.

The dream of the Semantic Web is to develop one ontology expressed in one language potentially covering everything that exists on the web. Berners-Lee [2] writes: "The Semantic Web is what we will get if we perform the same globalization process to Knowledge Representation that the Web initially did to Hypertext. We remove the centralized concepts of absolute truth, total knowledge, and total provability, and see what we can do with limited knowledge." As Noy and McGuinness [37] note, there are several excellent reasons for developing ontologies: to share a common understanding of the structure of information among people; to enable reuse of domain knowledge; to make domain assumptions explicit; to separate domain knowledge from the operational knowledge; and to analyse domain knowledge. The question is whether the ontologies as we know them from the current Semantic Web are the best tools to achieve these goals.

In reality, the picture is not so beautiful: there are several ontologies in several languages covering partly overlapping subdomains of the web. And the formalisms encounter several kinds of integration problems, including structural heterogeneity, semantic heterogeneity, inconsistency and redundancy problems [47]. Shirky [42] summarizes the state of the art as follows: "The Semantic Web, with its neat ontologies and its syllogistic logic, is a nice vision. However, like many visions that project future benefits but ignore present costs, it requires too much coordination and too much energy to effect in the real world, where deductive logic is less effective and shared world view is harder to create than we often want to admit."

One may even question whether we really need the meta-data provided by the Semantic Web. Uschold [46] points out that sometimes information integration works anyway, for example in web shopping agents. He writes that "[s]hopping agents can work even if there is no automatic processing of semantics; they can work without any formal representations of semantics; they can even work with no explicit representations of semantics at all. The key to enabling shopping agents to automatically use web content is that the meaning of the web content that the agents are expected to encounter can be determined by the human programmers who hardwire it into the web application software." The reason this can be done is that the terminology involved in web shopping application is comparatively limited and free from ambiguities.

2 The Semantic Web is not semantic

My main point of criticism is that the Semantic Web is not very semantic. At best, it is ontological. From a philosopher's point of view, it is not even ontological, since the formalisms exploited only provide a partial description of what a metaphysician would call an ontology. Berners-Lee, Hendler and Lassila [3] have the following comment: "Artificial-intelligence and Web researchers have co-opted the term for their own jargon, and for them an ontology is a document or file that formally defines the relations among terms. The most typical kind of ontology for the Web has a taxonomy and a set of inference rules." Even if we grant that the Semantic Web is ontological, it contains a number of competing ontologies that make information integration very problematic. Furthermore, the methodology of the Semantic Web puts too much emphasis on symbolic structures. This will be the topic of next section.

Why do I say that the Semantic Web is not semantic? Let us consider what can be expressed in, for example OWL (as far as I understand, other tools for the Semantic Web have similar properties). McGuinness and Hamelen [28] write that on top of RDF descriptions "OWL adds more vocabulary for describing properties and classes: among others, relations between classes (for example disjointness), cardinality (for example "exactly one"), equality, richer typing of properties, characteristics of properties (for example symmetry and transitivity), and enumerated classes." These are admittedly important semantic notions. But they are exactly what is to be expected of a language that has the expressivity of first order logic and that defines all concepts (properties and relations) in terms of *sets* of objects.

However, *there is much more to the semantics of concepts*. It is an unfortunate dogma of computer science in general, and the Semantic Web in particular, that all semantic contents are reducible to first order logic or to set theory. Berners-Lee, Hendler and Lassila [3] claim that "[f]ortunately, a large majority of the information we want to express is along the lines of 'a hex-head bolt is a type of machine bolt'." Unfortunately, this is not true. If one considers how humans handle concepts, the class relation structures of the Semantic Web capture only a minute part of our information about concepts. For instance, we often categorize objects according to the *similarity* between the objects [11, 14]. And similarity is not a notion that can be expressed in a natural way in a web ontology language.

Along the same lines, Shirky [42] declares that "the Semantic Web is a machine for creating syllogisms." He concludes that: "This is the promise of the Semantic Web – it will improve all the areas of your life where you currently use syllogisms. Which is to say, almost nowhere." He adds, somewhat sarcastically: "The people working on the Semantic Web greatly overestimate the value of deductive reasoning (a persistent theme in Artificial Intelligence projects generally.) The great popularizer of this error was Arthur Conan Doyle, whose Sherlock Holmes stories have done more damage to people's understanding of human intelligence than anyone other than Rene Descartes."

In my book *Conceptual Spaces* [11], I contrast three basic methodologies within the cognitive sciences for representing information: the symbolic, the associationist and the conceptual. In the symbolic approach, cognition is seen as essentially being *computation* involving symbol manipulation. The second approach is associationism, where *associations* between different kinds of information elements carry the main burden of representation. Connectionism is a special case of associationism that models associations using artificial neuron networks. The Semantic Web builds almost entirely on the symbolic methodology. The core of this paper will be to outline the conceptual approach to representations (sections 4-9). I shall argue that if we want to build real content into a system, one should rely on the conceptual methodology.

A remarkable feature of human thinking is our ability to *combine concepts* and, in particular, to *understand* new combinations of concepts [16]. Nobody has problems grasping the meaning of combinations like *pink elephant*, *striped apple* and *cubic soap-bubble*, even if one never will encounter any object with these properties. In all kinds of web applications, for example query answering systems, inputs in form of combinations of concepts are ubiquitous. Consequently, an important criterion for a successful computational model of the semantics of concepts is that it should be able to handle combinations of concepts.

In classical logic and in the Semantic Web, combinations of concepts are expressed by conjunctions of properties. This means that the reference of the combination of two concepts is taken to be the *intersection* of the classes representing of the two individual concepts. However, it turns out there are many everyday combinations of concepts that cannot be analysed in this simplistic manner. For example, *tall squirrel*, *honey bee*, *stone lion*, and *white Zinfandel* (which happens to be a rosé wine) cannot be analysed in terms of intersections of classes. In general, current symbolic methods have serious problems handling concept combinations in the way humans do.

3 Symbol grounding

Within general Artificial Intelligence there has been much discussion concerning the so-called "symbol grounding problem" [18]. The concern is how the symbolic expressions of first order logic or a programming language (including OWL and its relatives) can obtain any meaning that goes beyond the formal language itself – a meaning that is in some way "grounded" in the external world.

The symbol grounding problem shows up in relation to the Semantic Web in connection with competing ontologies. To see the point, consider how Noy and McGuinness [37] define this key concept: "An ontology is a formal explicit description in a domain of discourse (classes (sometimes called concepts)), properties of each concept describing various features and attributes of the concepts (slots (sometimes called roles or properties)) and restrictions on slots (facets (sometimes called role restrictions).” The key phrase here is “explicit description”. In languages such as DL, XML and OWL, descriptions are indeed explicit – this is seen as the very point of specifying an ontology.

One way to pinpoint the symbol grounding problem for the Semantic Web is to note that there is no general way to resolve a conflict between two ontologies, since all information is syntactic, that is, expressed in symbolic structures. This makes it difficult to match meanings of expressions in two different ontologies Even though various techniques for such a matching has been attempted [47], for example by mapping different taxonomies onto each other, they are all far from solving the general problem. Shirky [42] makes the following observation about the problem: “Any attempt at a global ontology is doomed to fail, because meta-data describes a worldview. The designers of the Soviet library's cataloging system were making an assertion about the world when they made the first category of books "Works of the classical authors of Marxism-Leninism." Melvyl Dewey was making an assertion about the world when he lumped all books about non-Christian religions into a single category, listed last among books about religion. It is not possible to neatly map these two systems onto one another, or onto other classification schemes – they describe different kinds of worlds.”

The source of the problem is that each ontology (together with its terminology) functions like a free floating island of reeds – it has no anchor in reality. And the “meanings” of the ontological expressions don’t live on these islands. It does not help to tie the islands together. To be sure, in the folklore around the Semantic Web one finds several special translation mechanisms to help the transition from one ontology to another (see for example [47]). Such a translation is a painstaking enterprise, but it does not solve the symbol grounding problem – even if mutually connected, the ontological islands have no connection with reality.

Aiming for explicit description is one of the dogmas of the tradition of symbolic knowledge representations. I want to challenge that explicit symbolic characterizations of ontologies is the most efficient, let alone the most natural (in relation to human cognition) way to represent this kind of knowledge for use on the Semantic Web. Shirky [42] formulates the problem in the following way: “Descriptions of the Semantic Web exhibit an inversion of trivial and hard issues because the core goal does as well. The Semantic Web takes for granted that many important aspects of the world can be specified in an unambiguous and universally agreed-on fashion, then spends a great deal of time talking about the ideal XML formats for those descriptions. This puts the stress on the wrong part of the problem -- if the world were easy to describe, you could do it in Sanskrit.” The main purpose of this article is to propose an alternative methodology and an alternative representational format.

Shirky [42] presents another critical example: “Consider another description of the Semantic Web that similarly misconstrues the problem:

> Merging databases simply becomes a matter of recording in RDF
> somewhere that "Person Name" in your database is equivalent to
> "Name" in my database, and then throwing all of the information
> together and getting a processor to think about it.
> [http://infomesh.net/2001/swintro/]

No one who has ever dealt with merging databases would use the word 'simply'. If making a thesaurus of field names were all there was to it, there would be no need for the Semantic Web; this process would work today. Contrariwise, to adopt a Lewis Carroll-ism, the use of hand-waving around the actual problem – human names are not globally unique – masks the triviality of linking Name and Person Name. Is your "Person Name = John Smith" the same person as my "Name = John Q. Smith"? Who knows? Not the Semantic Web. The processor could "think" about this til the silicon smokes without arriving at an answer." To be sure, human minds don't work that way – they have non-symbolic representations of objects, individuals and concepts that help them solve questions of this kind.

The rather radical solution I propose to circumvent the representational limitations of the current Semantic Web is to abandon its heavy dependence on syntactically expressed ontologies and go for *real* semantics instead. The remainder of this article will give a programmatic outline of how a richer semantic structure can be obtained by applying conceptual spaces as an underlying representational format.

4 Conceptual spaces as a framework for semantics

Conceptual spaces represent information by *geometric* structures rather than by symbols. Information is represented by points (standing for individuals or objects), and regions (standing for properties and relations) in dimensional spaces. Many semantic structures, for example similarity relations, can be modelled in a natural way by exploiting *distances* in the space. I call my way of representing information the *conceptual* form because I believe that the essential aspects of concept representation are best described using this approach. The framework presented here follows the theory put forward in my recent book *Conceptual Spaces: The Geometry of Thought* [11].

A conceptual space consists of a number of *quality dimensions*. Examples of such dimensions are: colour, pitch, temperature, time, weight, size and the three ordinary spatial dimensions. These examples are closely connected to what is produced by our sensory receptors. However, there are also quality dimensions that are of an abstract non-sensory character. For example in [12], the analysis is extended to functional concepts.

The primary role of the dimensions is to represent various "qualities" of objects in different *domains*. Since the notion of a domain is central to my analysis, it should be given a more precise meaning. To do this, I rely on the notions of separable and integral dimensions taken from cognitive psychology [13, 26, 30]. Certain quality dimensions are *integral* in the sense that one cannot assign an object a value on one dimension without giving it a value on the other. For example, an object cannot be given a hue, without also giving it a brightness value. Or the pitch of a sound always goes along with a loudness. Dimensions that are not integral are said to be *separable*, as for example the size and hue dimensions. Using this distinction, the notion of a *domain* can now be defined as a set of integral dimensions that are separable from all other dimensions. As will be seen below, the notion of a domain is central for the reconstruction of ontologies.

The domains form the framework used to assign *properties* to objects and to specify *relations* between them (see next section). The dimensions are taken to be independent of symbolic representations in the sense that we can represent the qualities of objects, for example by vectors, without presuming an explicit language in which these qualities are expressed.

The notion of a dimension should be understood literally. It is assumed that each of the quality dimensions is endowed with certain *topological* or *geometric* structures. As a first example, take the dimension of *time*. In science, time is a one-dimensional structure that is isomorphic to the line of real numbers. If *now* is seen as the zero point on the line, the future corresponds to the infinite positive real line and the past to the infinite negative line.

If it is assumed that the dimensions have a metric, one can talk about distances in the conceptual space. Such distances represent degrees of *similarity* between the objects represented in the space. Hence, conceptual spaces are suitable for representing different kinds of similarity relations.

A paradigmatic example of a domain involves *colour*. In brief, our cognitive representation of colour can be described by three dimensions. The first dimension is *hue*, which is represented by the familiar *colour circle*. The topological structure of this dimension is thus different from the quality dimensions representing time or weight which are isomorphic to the real line.

The second psychological dimension of colour is *saturation*, which ranges from grey (zero colour intensity) to increasingly greater intensities. This dimension is isomorphic to an interval of the real line. The third dimension is *brightness*, which varies from white to black and is thus a linear dimension with end points. Together, these three dimensions, one with circular structure and two with linear, constitute the colour domain which is a subspace of our perceptual conceptual space.

This domain is often illustrated by the so-called *colour spindle* (see figure 1). Brightness is shown on the vertical axis. Saturation is represented as the distance from the center of the spindle. Hue, finally, is represented by the positions along the perimeter of the central circle.

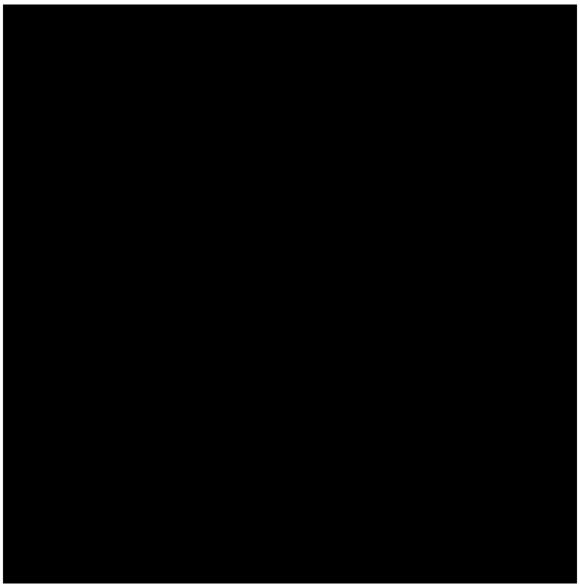


Figure 1: The colour spindle.

I cannot provide a complete list of the quality dimensions involved in the conceptual spaces of humans. Some of the dimensions seem to be innate and to some extent hardwired in our nervous system, as for example colour, pitch, and probably also ordinary space. Other dimensions are presumably learned. Learning new concepts often involves expanding one's conceptual space with new quality dimensions. Still other dimensions may be culturally dependent. Finally, some quality dimensions are introduced by science.

When it comes to implementing a knowledge representation system based on conceptual spaces, it is of course the programmer who decides the domains and their underlying structures, at least for the basic domains. This involves selecting the dimensions of the domain; specifying the topology or geometry of the domain; and, in the case of a metric space, specifying its metric. A rule of thumb for the metric is that integral dimensions combine by the Euclidean metric, while separable dimensions combine by the city-block (Minkowski) metric. In section 10, I shall discuss some of the computational aspects of this process.

5 Properties and concepts

The theory of conceptual spaces will first be used to provide a definition of a *property*. I propose the following criterion [6, 8] where the geometric characteristics of the quality dimensions are utilized to introduce a spatial structure for properties:

Criterion P: A *property* is a convex region in some domain.

The motivation for the criterion is that if some objects which are located at v_1 and v_2 in relation to some quality dimension (or several dimensions) are both examples of the property C, then any object that is located between v_1 and v_2 on the quality dimension(s) will also be an example of C. Criterion P presumes that the notion of *betweenness* is meaningful for the relevant quality dimensions. This is, however, a rather weak assumption which demands very little of the underlying geometric structure.

Criterion P does not presume that one can identify sharp borders between properties; it can be applied also to *fuzzy* properties or properties that are defined by *probabilistic* criteria. This is an advantage over most existing web ontologies. What convexity requires is that if two object locations v_1 and v_2 both satisfy a certain membership criterion, for example, has a certain degree or probability of membership, then all objects between v_1 and v_2 also satisfy the criterion.

Most properties expressed by simple words in natural languages seem to be natural properties in the sense specified here. For instance, I conjecture that all *colour terms* in natural languages express natural properties with respect to the psychological representation of the three colour dimensions. It is well-known that different languages carve up the colour circle in different ways [4], but all carvings seems to be done in terms of convex sets [11, 43].

Properties, as defined by criterion P, form a special case of *concepts*. I define this distinction by saying that a property is based on a *single* domain, while a concept may be based on *several* domains. The distinction between properties and concepts has been obliterated in the symbolic as well as connectionist representations that have dominated the discussion in the cognitive sciences. In particular, both properties and concepts are represented by *predicates* in first order languages. In web ontology languages, matters are made even less semantic by making *classes* the central units (classes are just the extensions of properties and concepts). The predicates of a first order language correspond to several different grammatical categories in a natural language, most importantly those of adjectives, nouns and verbs. The main semantic difference between adjectives and nouns, on the one hand, is that adjectives like “red,” “tall,” and “round” normally refer to a single domain and thus represent properties, while nouns like “dog,” “apple” and “town” normally contain information about several domains and thus represent concepts. Verbs, on the other hand, are characterized by their temporal structure, that is, they essentially involve the time dimension.

Let us now focus on the differences between single-domain properties and multi-domain concepts. As a paradigm example of a concept that is represented in several domains, consider “apple” (compare [45]). The first problem when representing a concept is to decide which are the relevant domains. When we encounter apples as children, the first domains that we learn about are presumably colour, shape, texture and taste. Later, we learn about apples as (biological) fruits, about their nutritional value, and possibly about some further dimensions. It should be noted that I do not require that a concept should be associated with a closed set of domains. On the contrary, this set may be expanded as one learns about further aspects of a concept.

The next problem is to determine the geometric structure of the domains. Taste space can presumably be represented by the four dimensions sweet, sour, salty and bitter and the colour domain by hue, saturation and brightness. Other domains are trickier: it is difficult to say much about, for example, the topological structure of "fruit space." Some ideas about how "shape space" should be modelled have been discussed in, for example [5, 7, 8, 11, 27]. Textures could possibly be modelled using fractal theory [39]. Instead of giving a detailed presentation of the geometric structures of the different domains, let me represent the "apple" regions verbally as follows:

Domain	*Region*
Colour	Red-yellow-green
Shape	Roundish (cycloid)
Texture	Smooth
Taste	Regions of the sweet and sour dimensions
Fruit	Specification of seed structure, flesh and peel type, etc, according to principles of pomology
Nutrition	Values of sugar content, vitamins, fibres, etc

When several domains are involved in a representation, some principle for how the different domains are to be *weighed* together must be assumed. These weights influence the distances in the conceptual space. The relative weights of the domains depend on the *context* in which the concept is used. Hence, I assume that in addition to the regions associated with each domain, the concept representation contains information about the *prominence* of the different domains. This aspect of concepts is not represented in existing web ontologies. The prominence values of different domains determine which *associations* can be made and thus which *inferences* can be triggered by a particular use of a concept (see [9]). The prominence values can change with the context, and with the knowledge and interests of the user. For example, if you are eating an apple, its taste will be more prominent than if you are using an apple as a ball when playing with an infant, which would make the shape domain particularly prominent.

Concepts are not just bundles of properties. The proposed representation for a concept also includes an account of the *correlations* between the regions from different domains that are associated with the concept. In the "apple" example there is a very strong (positive) correlation between the sweetness in the taste domain and the sugar content in the nutrition domain and a weaker correlation between the colour red and a sweet taste.

These considerations of prominence and correlations motivate the following definition of concept representation:

Criterion C: A *concept* is represented as a set of convex regions in a number of domains together with a prominence assignment to the domains and information about how the regions in different domains are correlated.

6 Connections to prototype theory

Criterion P (and thereby also Criterion C) derives independent support from the *prototype theory* of categorization developed by Rosch and her collaborators (see, for example [22, 31, 40,41]). The main idea of prototype theory is that within a category of objects, like those instantiating a property or a concept, certain members are judged to be more representative of the category than others. For example robins are judged to be more representative of the category "bird" than are ravens, penguins and emus; and desk chairs are more typical instances of the category "chair" than rocking chairs, deck-chairs, and beanbag chairs. The most representative members of a category are called prototypical members.

In the classical Aristotelian theory of concepts [44], a concept is defined via a set of necessary and sufficient properties. To a large extent, web ontologies follow this ideal in their characterizations of concepts. Consequently, all instances of a classical concept have equal status. However, another part of prototype theory says that concepts show graded membership, determined by how representative the members are. It is well-known that some properties, such as "red" and "bald" have no sharp boundaries, and for these it is perhaps not surprising that one finds prototypicality effects. However, these effects have been found for most concepts including those with comparatively clear boundaries like "bird" and "chair."

When natural properties are defined as convex regions of a conceptual space, prototype effects are indeed to be expected. In a convex region, one can describe positions as being more or less central. In particular, if the space has a metric, one can calculate the centre of gravity of a region.

It is possible to argue in the converse direction too and show that if prototype theory is adopted, then the representation of properties as convex regions is to be expected, at least in metric spaces. Assume that some quality dimensions of a conceptual space S are given, for example the dimensions of colour space, and that we want to partition it into a number of categories, for example colour categories. If we start from a set of prototypes p_1, ..., p_n of the categories, for example the focal colours, then these should be the central points in the categories they represent. If we make the additional assumption that S is a metric space, the information about prototypes can be used to generate a categorization. In order to see this, assume that S is equipped with the Euclidean metric so that for every point p in the space one can measure the distance from p to each of the p_i's. If we now stipulate that p belongs to the same category as the closest prototype p_i, it can be shown that this rule will generate a partitioning of the space – the so-called *Voronoi tessellation*. An illustration of the Voronoi tessellation is given in figure 2.

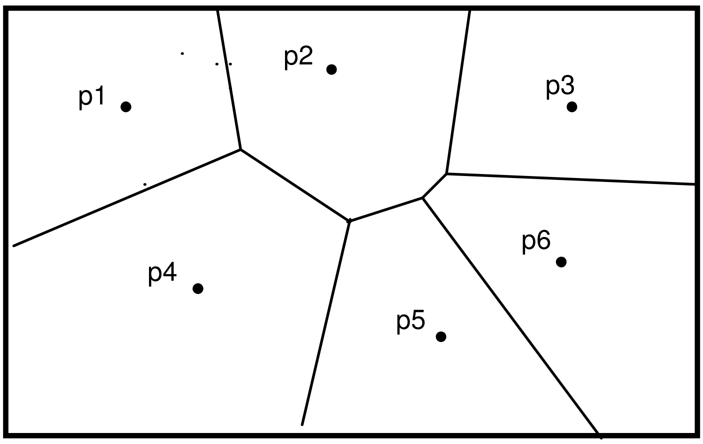


Figure 2. Voronoi tessellation of the plane into convex sets.

A crucial property of the Voronoi partitioning of a conceptual space is that the Voronoi tessellation based on an Euclidean metric always results in a partitioning of the space into convex regions (see [38]).

Thus, assuming that a Euclidean metric is defined on the subspace that is subject to categorization, a set of prototypes will by this method generate a unique partitioning of the subspace into convex regions. The upshot is that there is an intimate link between prototype theory and Criterion P. Furthermore, the metric is an obvious candidate for a measure of similarity between different objects. In this way, the Voronoi tessellation provides a constructive geometric answer to how a similarity measure together with a set of prototypes determine a set of categories. As will be seen in the following section, this way of generating categories can replace much of the work done on the Semantic Web by symbolic tools such as OWL.

In the analysis of concepts presented here, I have tried to bring in elements from other theories in psychology and linguistics. The kind of representation proposed in Criterion C is on the surface similar to *frames* with slots for different *features* (sometimes called slots, attributes or roles, see for example [37]) that have been very popular within cognitive science as well as linguistics and computer science. My definition is richer since a representation based on conceptual spaces will allow me to talk about concepts being *similar* to each other and about objects being more or less *central* representatives of a concept. My model can be seen as combining frames with prototype theory, although the geometry of the domains will make possible inferences that cannot be made in either frame theory or prototype theory (see [7]).

The main difference between the earlier theories and the conceptual spaces is that I put greater emphasis on the geometric structure of the concept representations. For example, features in frames are often represented in a symbolic form. As will be seen in sections 8 and 9, the geometric structures are essential for the modelling of concept combinations. The main advantage of the geometric information is that it will make more *inferences* possible – inferences that are based on similarity, context information and correlations between dimensions. These kinds of inferences are difficult to model in a purely symbolic framework.

The notion of a concept defined here has several similarities with the *image schemas* as studied in cognitive linguistics by Lakoff [22], Langacker [23] and others (see [10]). Even though their representations are often pictorial, they are, in general, not careful to specify the geometric structures of the domains that underlie the image schemas. Their account of image schemas does not generate machine-usable content, which is a necessary requirement for the Semantic Web. Holmqvist [20] has developed a computational-friendly representation that combines image schemas with some aspects of conceptual spaces. Many of his constructions could be applied also for the Semantic Web.

7 Conceptual spaces as a tool for the Semantic Web

Conceptual spaces can be seen as a representational level that serves as an anchoring mechanism *between* symbolic language and reality. Via representations of sensory dimensions, part of a conceptual space is tied to the “real world”. As explained in [11], other domains can then be added by metaphorical and higher level extensions. The connection between a conceptual space and a symbolic description is obtained by mapping name symbols onto *points* in the space and the concept (property and relation) symbols onto *regions* of the space (this will be explained further below). In this way the symbolic expression are anchored in a conceptual space, which in turn is partly “grounded” in reality.

I now want to outline how the representational format of conceptual spaces can be exploited to generate the kind of semantic structures that are needed for a Semantic Web worthy of its name.

The most important benefit of putting conceptual spaces as a representational level “below” the symbolic level is that the *concept hierarchies* (taxonomies) that are required for the symbolic structures can be generated almost for free from the conceptual spaces. The validity of “a robin is a bird” will *emerge* from the fact that the region of the conceptual space representing robins is a subregion of the one representing birds. In brief, if category prototypes and a metric domain are specified, a Voronoi tessellation can be computed that is sufficient to generate what is needed for an ontology in the sense specified above [37].

Identities of concepts are also immediate, once the mapping between predicate symbols and regions is established. For example, "vixens are identical with female foxes" will follow from the fact that "vixens" and "female foxes" are represented by the same region of the conceptual space.

Identities of names are handled in a similar way. Each name is mapped onto a point (vector) in the space. Hence, if two names (such as John Smith and John Q. Smith in Shirky's example above) are mapped onto the same point, they are identical in meaning. Matters become more complicated if names are mapped onto partial vectors, where the values for some dimensions may be missing. However, I will not pursue this theme here.

Property characteristics such as transitivity and symmetry also emerge from the representational format of the conceptual spaces. For example the fact that comparative relations like "earlier than" are *transitive* follows from the linear structure of the time dimension and is thus an intrinsic feature of this relation Similarly, it follows that everything that is green is coloured, since "green" refers to a region of the colour domain. Further, a *property constraint* such as that nothing is both red and green (all over) is immediate in the colour domain of the conceptual space, since these words refer to disjoint regions of the colour domain.

These aspects of ontologies of the type currently used for the Semantic Web can be generated from the *domain* structures of the conceptual space. To put it bluntly, once the conceptual structure of an informational system has been specified, the semantic information that OWL and other meta-data languages are supposed to add becomes redundant.

Another important aspect is that once this domain structure has been specified to allow representations of concept inclusion, property identity and name identity in the way outlined above, then there is no need for an inference engine on the symbolic level. The job of the symbolic reasoner is taken over by (non-symbolic) calculations on the conceptual level. As will be argued in section 10, these calculations are of a completely different nature compared to what is standard for symbolic inference engines.

However, more than taxonomies of concepts can be used to generate inferences. The great strength of using conceptual spaces is that they can be utilised for representing many other aspects of semantics as well. In particular, the information about *similarity* that comes with the conceptual spaces can be used for various forms of approximate reasoning on the Semantic Web. For example, if I am searching on the Web for a particular wine that I want to buy and the wine is not available, then, with the aid of the similarity information provided by the domain structures of a conceptual space, I can ask for a wine that is similar to the one I am looking for. Since similarity is dependent on which dimensions are focused on, the system should ask me for what dimensions of the wine that are most important before it determines which wines are the most similar. If we stick to the symbolic representational format for the Semantic Web, this kind of reasoning about similarities is well nigh impossible (unless the dimensional structure I built into the symbolic representation – but this would be putting the cart before the horse).

Another strong side of using conceptual spaces is that one obtains a much richer way of representing concept combinations. This will be the topic of next section. In [11], chapter 5, I also argue that *metaphors* and *metonymies*, to a large extent, can be understood using the conceptual framework. These are areas that are longstanding enigmas for the symbolic tradition. However, space does not allow me to develop these issues here.

Of course, the proposed methodology for generating semantic content is not without problems. Since so much is delegated to the structure of the domains, the questions of how to identify and describe domains become central. Will we not end up with the same problems as for symbolic ontologies? How should one, for instance, handle the situation if there are *competing* domain descriptions in two web applications? The beginning of an answer is that there is in general more agreement on how to define a domain than how to define an ontology for the existing Semantic Web. Even if there are competing domains, their structure can be compared via their geometric and topological properties (just as is happening in metaphorical mappings). To give an analogy from the philosophy of science: if Newtonian physics and relativity theory are compared as axiomatic theories (in some suitable symbolic formulation), the theories will be incompatible (Kuhn even says incommensurable). Most physicists don't care: if we view them as (dynamic) geometric structures, Newtonian mechanics will turn out to be just a limiting case of relativity theory.

8 Concept combinations

As was argued in section 2, the handling of concept combination is a central problem area for the Semantic Web, in particular for all kinds of query systems. Suppose that I am searching the Web for plants to my garden and in doing so I enter different combinations of words to my search engine. To just give one example, consider the use of "large" in combination like "large violet", "large flower" and "large garden plant". A large violet is not a large flower, let alone a large garden plant. Consequently, I would be disappointed if the search engine returned information about large violets, if I entered "large garden plant." This simple example shows that giving an encompassing definition of "large" will not work.

In the symbolic approach, combinations of concepts are traditionally expressed by conjunctions of predicates, modelled by intersections of classes. As was mentioned above, there are many everyday combinations of concepts that cannot be analyzed in this simplistic manner. Furthermore, as Hampton [17] points out, taking intersections leads to the prediction that if something is not a D, then it is not a D-which-is-a-C. But subjects tend to deny that a screwdriver is a weapon, but in general affirm that a screwdriver is a weapon-which-is-a-tool. This kind of non-monotonicity is ubiquitous in concept combinations. Similarly, "white Zinfandel" is not a sub-category of "white wine" (it is a rosé wine) and "white wine" is not a sub-category of "white objects" (it is light yellow).

In this section, I want to show that Criterion C has the potential to handle combinations of concepts. Suppose one wants to search the Web for the combination XY of two concepts X and Y, for example "large flowers" for my garden, where each concept is represented as a set of regions in a number of domains together with a prominence assignment to the domains according to the proposal in the previous section. Note that in the linguistic expressions for the combination, the order of X and Y is important: in English the word for X is taken to be a *modifier* of Y. Thus, *red brick* is a kind of brick, while *brick red* denotes a particular shade of red.

The most common case of concept combination is when X is a property (normally expressed by an adjective) and Y is a concept (expressed by a noun). But there are also cases where both concepts are multi-domain concepts, normally expressed by nouns, for example, *iron cast* that can be contrasted with *cast iron*. Noun-noun combinations have been studied in some detail in, among others, [33, 34, 48].

As a first approximation, the general rule for the combination XY of two concepts X and Y that I propose is that the region for some domain of the modifier X *replaces* the corresponding region for Y. However, it will turn out to be necessary to account for further factors governing concept formation. Consequently, this rule will be amended below.

Let us look at how the proposed rule applies to property-concept combinations. To give a paradigmatic example, *green apple* denotes the concept where the colour region of *apple* (which was illustrated as "red-yellow-green" in the frame above) is replaced by the more specific green region. In some cases, Y does not specify any value for the domain of X, in which case the X-region is simply added to the corresponding domain of Y. For example, the representation of *book* may not include any specified region of the colour domain, so in *yellow book* the yellow region of the colour domain is added to *book*.

If the region of X is *compatible* with the corresponding region of Y, like in the two examples above, the result of combining X and Y can be described as the *intersection* of the concepts, as was proposed for the classical logical theory. However, if the regions are incompatible, as in *pink elephant*, the region for X *overrules* that of Y (which, in the case of *elephant*, presumably is the grey region of the colour domain). In such a case, X "revises" Y and XY cannot be described in terms of intersections. Such revisions will result in *non-monotonic* effects of the contents of the concepts (see [9]).

Even if the region for X is not strictly incompatible with the region of Y, modifying Y by X may still lead to revisions of Y because of the *correlations* between domains that are also part of the concept representation proposed in the previous section. For example, in *brown apple*, modifying the colour domain to the brown region may lead to a modification of the texture domain from smooth to shrivelled ([43], p. 523) since there is a strong correlation among apples between being brown and being shrivelled. Similarly, in *wooden spoon*, the size region of the spoon will be changed from small to large when the material of the spoon is specified as wood (Medin and Shoben 1988).

In some cases, the regions of X may *block* some of the most prominent domains of Y, leading to rather drastic changes of the concept Y. For example in *stone lion*, the representation of *stone* includes the property "non-living" which is presumed by many of the domains of *lion*. These domains, like colour, sound, habitat, behaviour, etc., can thus not be assigned any region at all. By large, the only domain of *lion* that is compatible with *stone* is the shape domain. Consequently, the meaning of *stone lion* is an object made of stone that has the shape of a lion.

Holland et al. ([19], chapter 4.2) present a computer program for concept formation called PI and give some examples of how the program handles concept combination. Even though the system is rule-based, and thus essentially a symbolic system, it uses a frame structure with different "slots" (domains). Since representations in PI are based on (default or absolute) rules, it is difficult to see how a geometric structure could be modelled. The program will also have problems in situations when the rules defining the concepts to be combined are in conflict with one another as in the *stone lion* example. However, an advantage of their approach is that the rule-generating mechanisms take into account expectations about the *variability* of concepts. This will be the topic of next section.

9 The effect of contrast classes

To further motivate the geometric approach to concept combination, I now turn to a kind of combinations that cannot be handled properly by the symbolic approach. The starting point is that, for some concepts, the meaning of the concept is often determined by the context in which it occurs. Since these phenomena are not accounted for by the general rule for concept combination that was proposed above, this rule must be amended.

A typical example of context effects is that some properties cannot be defined independently of other properties. One simple example is the property of being *tall*. This property is connected to the height dimension, but cannot be identified with a region in this dimension. To see the difficulty, note that a chihuahua is a dog, but a tall chihuahua is not a tall dog. Thus "tall" cannot be identified with a set of tall objects. This property presumes some *contrast class* given by some other property, since things are not tall in themselves but only in relation to some given class of things. Tallness itself is determined with the aid of the height dimension. For a given contrast class Y, say the class of dogs, the region H(Y) of possible heights of the objects in Y can be determined. An object can then be said to be a tall Y if it belongs to the "upper" part of the region H(Y). Technically, the property "tall" can be defined as a class of regions: for each contrast class Y to which the height dimension is applicable, "tall" corresponds to the region of the height dimension which is the upper part of H(Y).

The same mechanism can be applied to a number of other properties. It can sometimes result in rather odd effects. For example, the same stream of tap water (with a given temperature x) can be described as "hot" if seen as water in general, but as "cold" if seen as bath water. The reason is that in the first case the region to which "hot" is applied is the full range of water temperatures, while in the latter case it is only the limited interval of bath water temperatures (see figure 3).

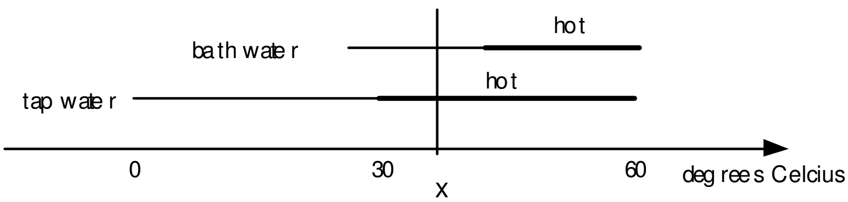


Figure 3. The meaning of "hot" in two different contrast classes.

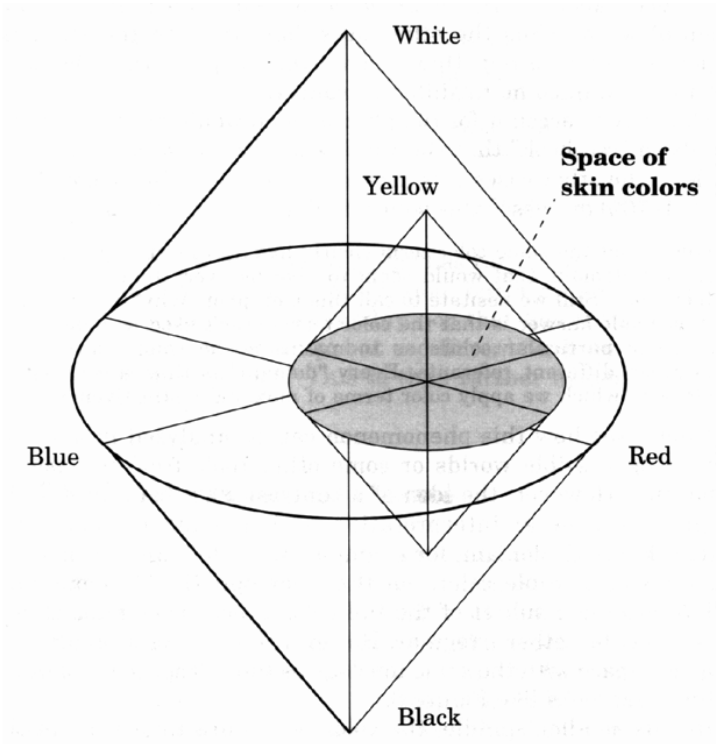
The effects of contrast classes also appear in many other situations. Consider the seemingly innocent concept *red* [15]. In the *Advanced Learner's Dictionary of Current English*, it is defined as "of the colour of fresh blood, rubies, human lips, the tongue, maple leaves in the autumn, post-office pillar boxes in Gt. Brit." This definition fits very well with letting *red* correspond to the standard region of the colour space. Now consider *red* in the following combinations:

- Red book
- Red wine
- Red hair
- Red skin
- Red soil
- Redwood

In the first example, *red* corresponds to the dictionary definition, and it can be combined with *book* in a straightforward extensional way that is expressed by a conjunction of predicates in first order logic (that is, intersecting the classes). In contrast, *red* would denote *purple* when predicated of wine, *copper* when used about hair, *tawny* when of skin, *ochre* when of soil and *pinkish brown* when of wood. How can we then explain that the same word is used in so many different contexts?

I don't see how this phenomenon can be analysed in a uniform way using a frame-based model or any other symbolic model and I would challenge proponents of these theories to come up with a solution that would work for the Semantic Web. Here I want to show how the idea that *a contrast class determines a domain* can quite easily be given a general interpretation with the aid of conceptual spaces. For each contrast class, for example skin colour, one can map out the possible colours on the colour spindle. This mapping will determine a subset of the full colour space. Now, if the subset is *completed to a space with the same geometry* as the full colour space, one obtains a picture that looks like figure 4.

In this smaller spindle, the colour words are then used in the same way as in the full space, even if the hues of the colour in the smaller space don't match the hues of the complete space. Thus, "white" is used about the lightest forms of skin, even though white skin is pinkish, “black” refers to the darkest form of skin, even though black skin is brown, etc. Note that the set of possible skin colours will not cover the entire small spindle, but



certain skin colour regions will be empty. There are for example no green people (but one can become green of envy or of sickness).

Figure 4: The subspace of skin colours embedded in the full colour spindle.

Given this way of handling contrast classes, I can now formulate a more precise version of the general rule for concept combination that was proposed above. The additional consideration is that the concept Y in the combination XY determines a contrast class. This contrast class may then *modify* the domains to which the concept X is applied as is illustrated in the examples above. The final proposal thus becomes:

The combination XY of two concepts X and Y is determined by letting the regions for the domains of X, confined to the contrast class defined by Y, replace the corresponding regions for Y.

In the last two sections, I have presented a geometric model of concept combination based on conceptual spaces. I have not strived at developing the model in a full formalism, mainly because for most natural concepts it is very difficult, at this stage, to identify the relevant geometric structures of the domains. The model should rather be seen as setting up a programme for future analyses of concepts. Nevertheless, my ambition has been to show that the proposed model can handle some of the cases that have been problematic for symbolic models of concept combination. It should be obvious that any improvements along these lines have immediate consequences for the quality of the Semantic Web. My model reminds of frames, but it strongly exploits the geometric structure of concept representations. Among other things, this structure makes it possible to talk about relative distances between concepts and to account for the main features of prototype theory.

10 Computational issues

The change of representational format that I am proposing will also have consequences for the programming methods that should be employed. The symbolic approach to information representation is intimately connected with the classical view of computation. On this view, computations are defined by the Turing machine paradigm. If we look at the methods used in the "symbol crunching" of traditional AI, a clearly dominating feature of the algorithms is that they implement some form of *rule following*. The rules can be logical axioms as in an automated theorem prover, they can be syntactic rules as in a parsing program, a rule formulated in OWL, or they can be of a cognitively more general type as in the ACT* [1] and SOAR [36] architectures. Since the rules often are applied iteratively, methods for handling *recursive procedures* become central elements of computer programs within the symbolic tradition. As a matter of fact, a mainstream position in mathematical logic and (Chomskian) linguistics is that the proper theory of inference is *nothing but* recursion theory.

When theories of theorem proving or sentence parsing are implemented in a computer, the program must be able to efficiently handle *tree-like structures*. This, in turn, entails that the computational methods require some smart heuristics for searching the tree structures. Consequently, using symbolic representations will lead to a focus on searching methods.

Turning next to computations with conceptual spaces, their basic representational elements are *points*. Mathematically, points in dimensional spaces can be seen as *vectors*. Thus, calculations on the conceptual level to a large extent involve *vector calculations*, using matrix multiplications, inner products, etc. The geometric properties of the vectors confer their basic representational capacities.

For example, the learning algorithms that are appropriate for representations on the conceptual level heavily exploit the vectorial representation. By using the distances provided in conceptual spaces, different numerical threshold criteria can be used in classification tasks (see, for example, Langley 1996, chapter 3, for a presentation of such criteria). In particular, linear and spherical threshold units have been studied. To classify an object using a linear threshold one multiplies the dimension values of the vector representing the object by the weights of the dimensions, sums the products, and checks whether the result exceeds the given threshold. This corresponds to using a (hyper-)plane in the conceptual space to classify the objects. The Voronoi rule is an example of such a linear threshold criterion. In a spherical threshold criterion all objects falling within a certain distance from a given point x are classified in the same category as x. This corresponds to using (hyper-)spheres in the space to sort the objects. Threshold criteria of these types obviously exploit the geometric structure of the conceptual space, particularly because they rely on distances.

One methodological feature that clearly distinguishes the conceptual level from the symbolic is that similarity plays a central role on the conceptual level. Similarity between objects or properties will be represented by distances in spaces. Such a notion of a distance is difficult to model in a natural way in a symbolic system. In a connectionist system, distances may appear as an emergent feature, but is difficult to represent directly in an artificial neuron network (unless it is built in, as in Kohonen systems).

Once distances between objects are represented, one can group objects according to the relative distances between them. In particular, *clusters* of objects are very useful when studying concept formation processes and generating taxonomies. Within the area of machine learning, there is a flourishing tradition concerning algorithms for identifying clusters (see, for example [21, 24, 32, 35]).

11 Conclusion

In this paper, my point of departure has been to identify some of the semantic limitations of the so-called Semantic Web. As a remedy, I have presented a research programme based on conceptual spaces. I have argued that by adopting this programme we obtain new tools to make the Semantic Web more semantic. In particular, information relating to similarity and the mechanisms of concept combination are important to increase the potency for inferences and query answering on the web.

The programme I propose involves a radically shift in focus of what meta-data should be added to the web. In the current Semantic Web, the information mainly concerns taxonomies and inference rules. If conceptual spaces are used as a foundational methodology, the focus will be on describing *domain structures*. This involves, above all, specifying the geometric and topological structure of the domains. Describing domains will require a different programming methodology compared to what exists in OWL and similar languages. The second step is to give information about how the resulting space is partitioned into concepts. I have shown that by using prototypes and Voronoi tessellations, this can be done in a computationally efficient way.

I have argued that once the relevant information about domain structure and prototypes becomes available, the taxonomies that have been in focus of the current Semantic Web will *emerge* from the domain structure and the symbolic inference mechanism will become more or less superfluous. The type of representing information suggested by conceptual spaces require computations involve vectors, using inferences based on similarities, rather than inference mechanisms based tree searching in a rule-based symbolic approach.

The main factor preventing a rapid advancement of different applications of conceptual spaces is the lack of knowledge about the relevant quality dimensions. Only for a few domain do we have sufficient psychophysical evidence to say that we have identified the geometric structure of the domains. For many of the domains that will be used in applications of the Semantic Web, this kind of information will not be available, but the programmer must rely on other methods to obtain the required domain structure. Here tools like Multi-Dimensional Scaling or Principal Component Analysis will become useful.

It is slightly discomforting to read that the philosopher John Locke already in 1690 formulated the problem of describing the structure of our semantic knowledge in his *Essay Concerning Human Understanding*: "[M]en are far enough from having agreed on the precise number of simple ideas or qualities belonging to any sort of things, signified by its name. Nor is it a wonder; since it requires much time, pains, and skill, strict inquiry, and long examination to find out what, and how many, those simple ideas are, which are constantly and inseparably united in nature, and are always to be found together in the same subject." ([25], book III, chapter VI, 30) Even though our knowledge has advanced a bit since then, we still face the same problems in the construction of the Semantic Web.

References

[1] J. R. Anderson. *The Architecture of Cognition*. Harvard University Press, Cambridge, MA, 1983.

[2] T. Berners-Lee. What the Semantic Web Can Represent. http://www.w3.org/DesignIssues/RDFnot.html, 1998.

[3] T. Berners-Lee, J. Hendler and O. Lassila. The Semantic Web. *Scientific American 284(5)*: 34-43, 2001.

[4] B. Berlin and P. Kay. *Basic Colour Terms: Their Universality and Evolution*. University of California Press, Berkeley, CA, 1969.

[5] I. Biederman. Recognition-by-Components: A Theory of Human Image Understanding. *Psychological Review 94*: 115-147, 1987.

[6] P. Gärdenfors. Induction, Conceptual Spaces and AI. *Philosophy of Science 57*: 78-95, 1990.

[7] P. Gärdenfors. Frameworks for Properties: Possible Worlds vs. Conceptual Spaces. In L. Haaparanta, M. Kusch and I. Niiniluoto, editors, *Language, Knowledge and Intentionality* (*Acta Philosophica Fennica,*

49), pages 383-407, 1990.
[8] P. Gärdenfors. A Geometric Model of Concept Formation. In S. Ohsuga et al., editors, *Information Modelling and Knowledge Bases III,* pages 1-16, IOS Press, Amsterdam, 1992.
[9] P. Gärdenfors. Concept Representations and Nonmonotonic Inferences. *Mathware & Soft Computing 3*: 17-31, 1996.
[10] P. Gärdenfors. Conceptual Spaces as a Framework for Cognitive Semantics. In A. Clark, editor, *Philosophy and Cognitive Science*, pages 159-180, Kluwer, Dordrecht, 1996.
[11] P. Gärdenfors. *Conceptual Spaces: The Geometry of Thought*. MIT Press, Cambridge, MA, 2000.
[12] P. Gärdenfors. Representing Actions and Functional Properties in Conceptual Spaces. Manuscript, to appear.
[13] W. R. Garner. *The Processing of Information and Structure*. Erlbaum, Potomac, MD, 1974.
[14] R. L. Goldstone. The Role of Similarity in Categorization: Providing a Groundwork. *Cognition 52*: 125-157, 1994.
[15] H. M. Halff, A. Ortony and R. C. Anderson. A Context-Sensitive Representation of Word Meanings. *Memory and Cognition 4*: 378-383, 1976.
[16] J. A. Hampton. Overextension of Conjunctive Concepts: Evidence for a Unitary Model of Concept Typicality and Class Inclusion. *Journal of Experimental Psychology: Learning, Memory and Cognition 14*: 12-32, 1988.
[17] J. A. Hampton. Conceptual Combination. In K. Lamberts and D. Shanks, editors, *Knowledge, Concepts and Categories*, pages 133-159, Psychology Press, Hove, 1997.
[18] S. Harnad. The Symbol Grounding Problem. *Physica D 42*, 335-346, 1990.
[19] J. H. Holland, K. J. Holyoak, R. E. Nisbett and P. R. Thagard. *Induction: Processes of Inference, Learning, and Discovery*. MIT Press, Cambridge, MA, 1995.
[20] K. Holmqvist. *Implementing Cognitive Semantics*, Lund University Cognitive Studies 17, Lund, 1993.
[21] N. Hulth and P. Grenholm. A Distributed Clustering Algorithm. *Lund University Cognitive Studies 74*, Lund, 1998.
[22] G. Lakoff. *Women, Fire, and Dangerous Things*. The University of Chicago Press, Chicago, IL, 1987.
[23] R. Langacker. *Foundations of Cognitive Grammar, Vol. 1*. Stanford University Press, Stanford, CA, 1987.
[24] P. Langley. *Elements of Machine Learning*. Morgan Kaufmann, San Francisco, CA, 1996.
[25] J. Locke. *An Essay Concerning Human Understanding*. Dover Publications, New York, NY, 1690/1959.
[26] W. T. Maddox. Perceptual and Decisional Separability. In G. F. Ashby, editor, *Multidimensional Models of Perception and Cognition*, pages 147-180, Lawrence Erlbaum Associates, Hillsdale, NJ, 1992.
[27] D. Marr, and H. K. Nishihara. Representation and Recognition of the Spatial Organization of Three-Dimensional Shapes. *Proceedings of the Royal Society in London, B 200*: 269-294, 1978.
[28] D. L. McGuinness and F. van Hamelen. OWL Web Ontology Language Overviews, W3C Recommendation 10 February 2004. http://www.w3.org/TR/2004/REC-owl-features-20040210/, 2004.
[29] D. L. Medin and E. J. Shoben. Context and Structure in Conceptual Combination. *Cognitive Psychology 20*: 158-190, 1988.
[30] R. D. Melara. The Concept of Perceptual Similarity: From Psychophysics to Cognitive Psychology. In D. Algom, editor, *Psychophysical Approaches to Cognition*, pages 303-388, Elsevier, Amsterdam, 1992.
[31] C. Mervis, and E. Rosch. Categorization of Natural Objects. *Annual Review of Psychology* 32: 89–115, 1981.
[32] R. S. Michalski and R. E. Stepp. Learning from Observation: Conceptual Clustering. In R. S. Michalski, J. G. Carbonell and T. M. Mitchell, editors, *Machine Learning, An Artificial Intelligence Approach*, pages 331-363, Morgan Kaufmann, Los Altos, CA, 1983.
[33] G. L. Murphy. Comprehending Complex Concepts. *Cognitive Science 12*: 529-562, 1988.
[34] G. L. Murphy. Noun Phrase Interpretation and Conceptual Combination. *Journal of Memory and Language 29*: 259-288, 1990.
[35] F. D. Murtagh. Cluster Analysis Using Proximities. In I. van Mechelen *et al*., editors, *Categories and Concepts: Theoretical Views and Data Analysis*, pages 225-245, Academic Press, London, 1993.
[36] A. Newell. *Unified Theories of Cognition*, Harvard University Press, Cambridge, MA, 1990.
[37] N. F. Noy and D. L. McGuinness. Ontology Development 101: A Guide to Creating Your First Ontology. *Stanford Knowledge Systems Laboratory Technical Report*, Stanford, CA, 2001.
[38] A. Okabe, B. Boots and K. Sugihara. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, John Wiley and Sons, New York, NY, 1992.
[39] A. P. Pentland. Perceptual Organization and the Representation of Natural Form. *Artificial Intelligence 28*: 293-331, 1986.
[40] E. Rosch. Cognitive Representations of Semantic Categories. *Journal of Experimental Psychology: General 104*: 192–233, 1975.

[41] E. Rosch. Prototype Classification and Logical Classification: the Two Systems. In E. Scholnik, editor, *New Trends in Cognitive Representation: Challenges to Piaget's Theory*, pages 73–86, Lawrence Erlbaum Associates, Hillsdale, NJ, 1978.
[42] C. Shirky. The Semantic Web, Syllogism, and Worldview. http://www.shirky.com/writings/semantic\-syllogism.html, 2003.
[43] L, Sivik and C. Taft. Colour Naming: A Mapping in the NCS of Common Colour Terms. *Scandinavian Journal of Psychology 35*: 144-164, 1994.
[44] E. E. Smith and D. L. Medin. *Categories and Concepts*, Harvard University Press, Cambridge, MA, 1981.
[45] E. E. Smith, D. N Osherson, L. J. Rips and M. Keane. Combining Prototypes: A Selective Modification Model. *Cognitive Science 12*: 485-527, 1988.
[46] M. Uschold. Where are the Semantics in the Semantic Web? *AI Magazine 24*[3]: 25-36, 2003.
[47] U. Visser. Ontology-Based Information Integration for the Semantic Web. http://ails04.cs.lth.se/Visserabstract.pdf, 2004.
[48] E. J. Wisniewski. Construal and Similarity in Conceptual Combination. *Journal of Memory and Language 35*: 434-453, 1996.