Cognitive Science 34 (2010) 222–254

ISSN: 0364-0213 print / 1551-6709 online
DOI: 10.1111/j.1551-6709.2009.01068.x

Strudel: A Corpus-Based Semantic Model Based on Properties and Types

Marco Baroni, Brian Murphy, Eduard Barbu, Massimo Poesio

Center for Mind Brain Sciences, University of Trento

Received 22 September 2008; received in revised form 11 July 2009; accepted 15 July 2009

Abstract

Computational models of meaning trained on naturally occurring text successfully model human performance on tasks involving simple similarity measures, but they characterize meaning in terms of undifferentiated bags of words or topical dimensions. This has led some to question their psychological plausibility (Murphy, 2002; Schunn, 1999). We present here a fully automatic method for extracting a structured and comprehensive set of concept descriptions directly from an English part-of-speech-tagged corpus. Concepts are characterized by weighted properties, enriched with concept–property types that approximate classical relations such as hypernymy and function. Our model outperforms comparable algorithms in cognitive tasks pertaining not only to concept-internal structures (discovering properties of concepts, grouping properties by property type) but also to inter-concept relations (clustering into superordinates), suggesting the empirical validity of the property-based approach.

Keywords: Corpus-based semantic models; Conceptual knowledge induction; Property-based concept representations

1. Introduction

Computational systems that induce facets of semantic representation from prepared or naturally occurring input are a valuable complement to experimental data, providing direct evidence about the informativeness of the input and the a priori structures that must be assumed for successful learning.

A particularly interesting class of computational methods is that of corpus-based semantic models (CSMs) that learn aspects of semantics from patterns of co-occurrence in

Correspondence should be sent to Marco Baroni, Centro Interdipartimentale Mente/Cervello, Università di Trento, corso Bettini 31, I-38068 Rovereto (TN), Italy. E-mail: marco.baroni@unitn.it

naturally occurring data, mostly in the form of linguistic corpora, that is, large and typically mixed collections of texts that have been produced for independent communicative purposes (e.g., collections of newspaper articles, book chapters, Web pages, conversation transcriptions, etc.). CSMs are interesting because, by operating on corpora, they must cope with problems of noise, skewed distributions and scarcity of explicit and coherent information that are presumably also faced by humans when acquiring language and conceptual knowledge. Moreover, CSMs can harvest semantic knowledge resources on a very large scale, far beyond what can be assembled by hand at reasonable costs. Pioneering CSMs such as HAL (Lund & Burgess, 1996) and LSA (Landauer & Dumais, 1997) and those that followed in their steps exploit the idea that semantically similar words occur in similar contexts, and thus represent the meaning of a word/concept by a numerical ''context signature'' that keeps track of the contexts in which the word appears in a corpus. Standard mathematical techniques can then be used to quantify semantic similarity by comparing the context signatures of different words/concepts.

The context signatures of typical CSMs are not interpretable as characterizations of the concepts in terms of their constituent properties, of the sort that are postulated in many fields of cognitive science. Consequently, CSMs might be very good at finding out that two concepts are similar, but they tell us little about the internal structure of concepts and, hence, why or how they are similar. As Murphy (2002, p. 429) puts it: ''Although *jugs* might be related to both *vinegar* and *bottles*, these relations are extremely different, and an overall similarity score does not represent these differences.'' In order to distinguish how *jugs* are related to *vinegar* from how they are related to *bottles*, one needs to know what are the properties of these concepts and the type of these properties: ''[S]ince one's concept of a jug, say, would include detailed information about its origins, parts, materials, functions and so on, the concept is more than sufficient to distinguish the meaning of *jugs* from that of *vinegar* and, for that matter, *bottles*.'' There is a wide consensus in cognitive science that the representation of concepts must include some form of decomposition into properties, and that these properties are organized according to how they relate to the concept (Fodor, 1998, is a notable atomistic exception). Such property-based organization has played a critical role in theorizing about concepts in psychology (McRae, de Sa, & Seidenberg, 1997; Murphy, 2002; Plaut & Shallice, 1993; Rosch, 1975; Vigliocco, Vinson, Lewis, & Garrett, 2004), neural science (Capitani, Laiacona, Mahon, & Caramazza, 2003; Martin & Chao, 2001), linguistics (Jackendoff, 1990; Pustejovsky, 1995) and artificial intelligence (Brachman & Schmolze, 1985; Norman, Rumelhart, & the LNR Research Group, 1975; Rogers & McClelland, 2004; Woods, 1975).

Here, we propose Strudel (structured dimension extraction and labeling), a new CSM that represents concepts in terms of *interpretable typed properties*, that is, it not only discovers that *dogs* are similar to *cats*, but it also finds out what are the salient properties of *dogs* (that they *are animals*, they *bark*, etc.), and it gives at least some indication of the types of relations instantiated by the *dog* concept with its properties (*being an animal* is a categorical property, *barking* a typical behavior, etc.). By producing concept descriptions in terms of weighted typed properties, Strudel brings corpus-based computational modeling of concept acquisition closer to work on concepts in the rest of cognitive science. We want to take up

the challenge of Gregory Murphy: ''Although current implementations [of CSMs] do not generally result in many interpretable dimensions, it is again possible that future versions will. If so, then we might be able to use LSA[-like] representations to derive more familiar conceptual representations of feature lists and schemata'' (Murphy, 2002, p. 430).

We will show that Strudel is producing overall plausible property-based concept characterizations (that are in part similar, in part different—in interesting ways—from those produced by subjects in property elicitation experiments), and that it can use such characterizations to achieve competitive performance in concept categorization, a classic task that requires measuring the similarity between concepts, thus suggesting that property-based characterizations derived with a computational approach are a viable means to represent concepts for similarity tasks.

The remainder of this paper is structured as follows. In Section 2, we briefly review the computational literature on concept acquisition. The Strudel algorithm is introduced in Section 3. Section 4 provides implementation details for Strudel and the CSMs we compared it against. Examples of the property-based concept descriptions generated by Strudel and the alternative CSMs are presented in Section 5. Section 6 shows that the most salient properties produced by Strudel for a range of concepts are significantly closer to those produced by humans, when compared with other CSMs, and discusses the differences between Strudel- and human-generated properties. Section 7 provides evidence that Strudel is grouping properties into meaningful general types, such as category or location. In Section 8, we show that Strudel outperforms comparable CSMs in a concept categorization task, and we analyze the superordinates that the algorithm discovered as a by-product of this task, as well as their hierarchical structure. We conclude in Section 9 by summarizing our current achievements and discussing directions for further work.

The paper is supplemented by online materials available from http://www.cogsci.rpi.edu/CSJarchive/Supplemental/index.html. This Supporting Information includes a detailed technical description of Strudel with input and output examples, code, the full trained model used in this article and data from the clustering experiments of Sections 7 and 8.

We briefly remark here that we will use the terms *meaning of a word* and *concept* interchangeably (on the relationship between the two notions, see Murphy, 2002, chapter 11). We are aware that, by making this potentially dangerous simplification, we gloss over important issues such as the existence of concepts that are not verbalized (or at least not lexicalized as single words), the possible distinction between ''linguistic'' and ''encyclopedic'' aspects of word meaning, as well as polysemy, homonymy, and synonymy. We think it is reasonable to first lay out and test the basic tenets of our approach, and to come back to these difficult problems in future research.

2. Computational models of concept induction

Corpus-based semantic models induce concepts/word meanings from large-scale, realistic linguistic input, that is, language corpora (see Sahlgren, 2006, for an overview). HAL (Lund & Burgess, 1996) and LSA (Landauer & Dumais, 1997) are among the oldest and most

well-known CSMs. These and many subsequent models have achieved impressive results in simulations of a variety of semantic knowledge tasks ranging from discovering synonyms (Landauer & Dumais, 1997) to modeling semantic priming (Lund, Burgess, & Atchley, 1995), categorization (Almuhareb, 2006; Laham, 1997), and language acquisition (Baroni, Lenci, & Onnis, 2007; Li, Burgess, & Lund, 2000). CSMs are also used to develop semantic resources for applications in natural language processing and related areas, where it would be extremely expensive and time consuming to produce resources of comparable scale by manual annotation (see, among others, Louwerse, Cai, Hu, Ventura, & Jeuniaux, 2006; Sahlgren, 2006).

Corpus-based semantic models adopt the general idea that similar words will be used in similar contexts and represent the meaning of a word/concept as a vector (ordered list) of values that summarize, in some way, the context signature of the word, that is, the set of contexts in which the word appears. Thanks to these vectors, words can be represented as points in a high-dimensional space, and the semantic distance between words is measured using standard geometric (or probabilistic) techniques, such as computing the cosine of the angle between two vectors.

Most CSMs can be seen as ''flat'' models of concepts, making the minimal stipulation that conceptual descriptions include information about the lexical neighbors of a word; such a description might include the information that there is a relationship between *cores* and *apples*—that is, the word *core* is one of the features of the lexical description of the word *apple*—but no claim is made that the type of relation is included in the description, and not all neighbors correspond to meaningful semantic properties of the concept: co-occurring words will also include collocations (*bad apple*) and generic terms of various sorts (among the words with the highest weighted co-occurrence with *apple* in the re-implementation of HAL we use in our laboratory we encounter: *variety*, *such*, *include*, *get*, *day*). When co-occurrence vectors undergo dimensionality reduction (a mathematical operation aiming at making the vectors more manageable while capturing generalized patterns of co-occurrence) as they do in LSA, the resulting features correspond to very broad, topical semantic domains (such as *traffic* and *food*; see Baroni & Lenci, 2008, for examples). These might be informative if the goal is extracting the gist of a text (Griffiths, Steyvers, & Tenenbaum, 2007; Steinberger, Poesio, Kabadjov, & Jezek, 2007) or measuring textual coherence (Foltz, Kintsch, & Landauer, 1998), but they are rather different from the classical concept properties studied in cognitive science (parts, functions, visual qualities, etc.).

For the experiments of this article, we re-implemented a ''classic'' CSM model that we call the singular value decomposition (SVD) model, inspired by those of Schütze (1997) and Rapp (2003, 2004). The latter model is currently the CSM with the best performance on the standard TOEFL synonym detection task (92.5% accuracy). Like HAL, our SVD model relies on a word-by-word matrix, rather than a LSA-like word-by-document matrix. However, the model is similar to LSA in that the original $m \times n$ word-by-word matrix (with m target and n context words) is reduced, using the singular value decomposition (SVD) dimensionality reduction technique, to a $m \times r$ word-by-weighted-left-singular-vector matrix (where r is much smaller than n). Note that SVD is not just an operation aimed at making the matrix more manageable. More importantly, the new dimensions are supposed to

emphasize patterns of correlations in the original columns while discarding noise (Landauer & Dumais, 1997). The new columns can be interpreted as ''latent'' dimensions that capture more robust similarity profiles than the original columns.

While HAL, LSA, and related models (including our SVD model) are based on co-occurrence in documents or text windows, one line of research on CSMs has been aiming towards a more linguistically aware definition of context. Starting from the seminal work by Grefenstette (1994), models have been developed that rely on parsed corpora to harvest linguistically interesting collocates, for example, objects, but not adjuncts of a verb (Lin, 1998; Curran & Moens, 2002). The intuition behind this approach is that sharing such contexts is a stronger cue of semantic similarity than simple window- or document-based co-occurrence. The recent study of Padó and Lapata (2007) shows how a CSM trained on contexts filtered by syntactic dependency relations outperforms comparable models in a variety of tasks. In the experiments below, we use a version of Padó and Lapata's CSM (the dependency vectors [DV] model) retrained on the same data Strudel was trained on.

A different line of research within CSMs aims at extracting semantic relations from corpora by using relation extraction methods developed in computer science and artificial intelligence for application domains such as information extraction and ontology population. Relation extraction methods (Girju, Badulescu, & Moldovan, 2006; Hearst, 1992, 1998; Pantel & Pennacchiotti, 2006, among many others) focus on learning one relation type at a time (e.g., finding the superordinate, or the parts of a concept). In Hearst's classic work, manually crafted lexico-syntactic patterns are used to harvest properties of the relevant type from corpora (e.g., the pattern *X such as Y* is likely to cue a super-to-subordinate relation), whereas in more recent work (e.g., Pantel & Pennacchiotti, 2006) the algorithm discovers the patterns associated with a relation from a list of example pairs (*car-vehicle*, *wolf-mammal*, etc.) to be searched in the corpus.

Recently, there have been attempts to use such techniques to extract complete concept descriptions (Almuhareb & Poesio, 2004; Cimiano & Wenderoth, 2005; Poesio & Almuhareb, 2005, 2008). Almuhareb and Poesio (2004) developed a totally unsupervised model that exploits surface cues to semantic relations (their subsequent work adopts a supervised approach). By taking inspiration from the AI literature on conceptual modeling, Almuhareb and Poesio assume that concepts can be described in terms of *attributes* that represent parts of entities and the general ''intrinsic'' dimensions of conceptualization among which entities will differ (e.g., *size*, *color*, and *shape* might be important attributes to characterize animals) and *values* that a concept might take for these attributes (*small*, *green*, and *round* are possible values of the attributes above). The dimensions of their CSM contain potential attributes and values, harvested with simple surface patterns. For the experiments below, we re-implemented Almuhareb and Poesio's unsupervised approach, called here the attribute value (AV) model.

A general problem with relation extraction algorithms is that, although they often produce high-quality results, their input must be customized (with separate manually crafted resources: patterns and/or example lists) for each relation type to be induced (but see Banko, Cafarella, Soderland, Broadhead, & Etzioni, 2007; Banko & Etzioni, 2008; Davidov & Rappoport, 2008a,b for some recent methods that discover and acquire multiple relation

types at once). Besides the laboriousness of this procedure, one has to postulate, a priori, the full set of relation types which are most appropriate for characterizing concepts, according to some existing theory. Minimally, this implies that an important aspect of semantic knowledge will be manually pre-encoded in the algorithm instead of being induced from the data.

Summarizing, the three CSMs we compare Strudel against represent three broad lines of CSM research: window-based co-occurrence approaches (the SVD model), dependency-parse-based models (the DV model), and pattern-based relation extraction (the AV model). We do not claim that these are the ''best'' models in the field, but they represent a set of current approaches that are closely related to ideas implemented in Strudel. We need to stress that, while we will show that Strudel outperforms these alternative CSMs on tasks that pertain to concept description and categorization, there are other, equally important aspects of semantic knowledge that might be better captured by other models. For example, we do not expect Strudel to perform well on tasks requiring the identification of topical domains, where latent dimension SVD-based models (and of course the topic models we are about to review) are likely to fare much better.

Two recent (families of) CSMs, that we did not re-implement, are moving, like us, toward a more flexible view of corpus-based semantics. Topic models (Blei, Ng, & Jordan, 2003; Griffiths et al., 2007; Hofmann, 2001) extract representations of concepts in terms of a latent distribution over ''topics.'' These models have shown good performance in classic semantic similarity tasks, as well as for extracting the gist of a document and detecting the right sense of a polysemous word in context. Moreover, thanks to their rigorous formalization as generative models, they can be integrated with other probabilistic methods, for example, Hidden Markov Models of word sequences, to discover syntactic and semantic information in parallel. Topic models are an extremely interesting development, aiming, like Strudel, to provide a more nuanced view of semantic similarity and concept descriptions. However, by focusing on topics, they are largely complementary to what we are trying to pursue here. We model concepts in terms of the typed properties that characterize them (*dogs* have *tails* as parts, *barking* as behavior), topic models characterize concepts in terms of the topics they belong to (*dogs* might belong to *zoology* as well as *family life*, etc.). Future research should pursue the integration of a probabilistic formulation of Strudel with topic models.

BEAGLE is another recently proposed CSM (Jones & Mewhort, 2007) that produces representations that contain at the same time semantic and syntactic information about words. Among other things, BEAGLE captures, in a completely knowledge-free way, a distinction between two basic types of similarity: taxonomic similarity (*car* and *boat*, *bus*), and associative similarity between concepts that tend to co-occur and thus likely belong to the same domain/frame (*car* and *driver*, *wheels*, *road*). BEAGLE could indeed detect the distinction between the *bottle–jug* and *bottle–vinegar* relations discussed by Murphy (see Section 1), the former being taxonomic, the latter associative. Still, there are many types of associative relations (we should also distinguish, e.g., between the relations instantiated by *bottle–vinegar*, *bottle–kitchen*, and *bottle–drink*), and the current version of BEAGLE is not well suited to capture this richness of types.

While our survey focused on CSMs, we should not forget that the earliest and most extensive computational tradition in concept learning is the one rooted in connectionism

(Rumelhart, McClelland, & the PDP Research Group, 1986). Neural networks have been applied to an impressive range of semantic tasks, through fine-tuned simulations of many aspects of conceptual cognition (McClelland & Rogers, 2003; McRae et al., 1997; Plaut, 1995; Plaut & Shallice, 1993; Rogers & McClelland, 2004; Rumelhart et al., 1986, among many others). On the one hand, these simulations typically pay much more attention to cognitive and biological plausibility than CSMs. On the other, their input is mostly made of small, hand-crafted lists of propositions (*cats have tails*, *bananas are yellow*)—a choice that might be due to efficiency issues with neural network architectures, but it is also motivated by the desire to zero in on how specific characteristics of the input impact learning (Rogers & McClelland, 2004). Connectionist models, then, do not need to cope with the problems of noise and highly skewed distributions found in naturally occurring data. For example, in the two billion word corpus described in Section 4, the proposition *cats have tails* occurs just one time, whereas *cats have staff* occurs six times, and *cats have access* three times. Moreover, by relying on manually crafted input, typical connectionist models do not scale up to realistically sized representations of human conceptual cognition. Despite these differences, CSMs and connectionism are by no means opposite paradigms. Neural networks, like CSMs, emphasize the role of learning from simple statistical/distributional cues, and the two approaches can and have been profitably combined (see, e.g., Moscoso del Prado & Sahlgren, 2002).

3. Strudel

Unlike the connectionist models and like other CSMs, Strudel (short for ''structured dimension extraction and labeling''—i.e., labeling of the extracted dimensions by type) induces semantic information from naturally occurring data without supervision and requiring a minimal amount of pre-encoded knowledge (part-of-speech [POS] tagging and lemmatization of the corpus, and a set of extraction templates defined over POS sequences). However, unlike traditional CSMs and like relation extraction algorithms, the dimensions of a Strudel semantic space are interpretable as properties, automatically annotated with information about the nature of the relation they instantiate. Unlike most relation extraction algorithms, Strudel does not start with a predefined set of relations, but it automatically identifies the most distinctive properties for each concept, thus providing useful evidence for what we might call the activity of ''empirical ontology''—identifying the set of relations and properties that are most characteristic of a concept.

The property extraction and typing algorithm of Strudel is based on three fundamental intuitions:

1. The patterns connecting concepts and properties in text can be captured by a restricted number of templates that define, at a very general level, which sequences of POS are possible concept–property connector patterns (we do not pre define specific connector patterns, we just specify high-level filters and induce the actual patterns from the data).

2. The *variety* of patterns connecting a concept and a potential property is a good indicator of the presence of a true semantic link (as opposed to simple collocational association): properties are hence scored based on the number of *distinct* patterns connecting them to a concept, rather than on the overall number of corpus co-occurrences.
3. While a single pattern connecting a property to a concept is ambiguous, the *distribution* of patterns connecting them provides an implicit characterization of the type of relation that exists between them.

Given a list of concepts and a corpus, Strudel builds structured representations of the concepts in two phases. First, it uses pattern filtering to identify and score potential properties of the concepts. In the second phase, Strudel assigns a type distribution to each scored concept–property pair by generalizing from the strings connecting them.

We now proceed to describe these phases in more detail. The Supporting Information supplements the description given here with a more technical introduction, the Strudel code, and the trained model used in our experiments.

3.1. Property extraction and scoring

Given a list of nominal concepts and a POS-tagged corpus, Strudel looks for nouns, adjectives, and verbs that occur near a concept. Only content words that are linked to it by a connector pattern that matches one of a limited set of templates are considered potential properties.

Strudel generalizes the idea of Hearst (1992) that simple surface patterns can be a cue to the presence of interesting semantic relations in natural text. Hearst was interested in specific relations, and thus harvested them using specific patterns. In our case, we are interested in building a general description of a concept in terms of its properties, and we cannot predict which relations will be salient for a certain concept. Thus, rather than focusing on specific patterns, we use a simple ''pattern grammar'' that filters out word sequences that would not make plausible connector patterns. The pattern grammar says, for example, that prepositions might cue an interesting relation between two nouns (however, it does not specify a list of specific prepositions of interest, like in Hearst's approach), whereas a sequence of a verb and a conjunction between the two nouns probably will not, and we thus weed out co-occurrences of nouns in this context.

The templates for nominal properties specify that the target and property must either be adjacent (the noun–noun compound case) or they must be connected by a (possibly complex) preposition, or a verb, or the possessive (*'s*), or a relative such as *whose*. Optional material, such as adjectives and articles, can occur in the connector pattern, whereas other categories, such as names and sentence boundaries, act as barriers blocking the template match. The template-matching component also performs basic pattern normalization by replacing all verbs and adjectives that are not in a ''keep list'' of 50 frequent verbs and 10 frequent adjectives with the corresponding POS tags; and it keeps track of the relative ordering of concept and candidate property (consider *onion with different layers* and *layer from an onion*). Table 1 presents examples of the extraction procedure for the concept *C onion* and the candidate property *P layer*.

Table 1
Examples of input and output of the Strudel pattern template component

Input	Output	Notes
Layer from an onion	P_from_a_C	*an* normalized to *a*
Layers in a red onion	P_in_a_ADJ_C	*red* mapped to *ADJ*
Onion with different layers	C_with_different_P	Frequent adj *different* preserved
Onions and with their layers	Ø	Conjunction blocks pattern extraction

Similar rules are applied to the extraction of adjective and verb properties. The whole template-matching component (presented in full in the online technical report) amounts to 15 rules, where each rule is implemented as a regular expression.

Concept–property pairs are then scored based on the number of distinct patterns that link them, ignoring the token frequency of the concept–property–pattern tuples. The intuition behind this approach is that a single, frequent concept–pattern–property tuple could simply be a fixed expression, or more in general a combination that is frequent for accidental reasons. On the other hand, if concept and property appear with many distinct patterns, that is, their relation is predicated in many different ways, it is more likely that they are connected by an inherent semantic link. For example, *year of the tiger* is much more frequent in our corpus than any pattern connecting *tail* and *tiger*. However, *year of the tiger*, because of its idiosyncratic nature and proper-noun-like usage, is the only attested pattern linking these two words (we do not find: *year of some tigers*, *tigers have years*, etc.). The relationship of *tigers* with *tails*, on the other hand, is expressed in a number of ways: *tail of the tiger*, *tail of a tiger*, *tigers have tails*, *tigers with tails*, etc. Pattern variety is a better cue to semantics than absolute frequency (although we only evaluated this claim qualitatively for now, and we postpone a systematic study of the effect of pattern diversity vs. frequency to future work).

More precisely, our score is based on the strength of the statistical association between concepts and properties sampled from the list of distinct tuples (akin to sampling concepts and properties from a dictionary of distinct longer strings rather than from a corpus). Association, measured by the log-likelihood ratio statistic (Dunning, 1993), is better than raw frequency as it weights down properties that might occur in a number of patterns simply in virtue of their generic nature (e.g., *year* and *time* that can occur with almost anything).

Given the four O_{ij} cells of the 2×2 contingency table for the (co-)occurrence of a concept and a property (where the i and j indices range over the rows and columns of the table) and the four E_{ij} cells of the corresponding table of expected occurrences under independence, we calculate the log-likelihood ratio score of the concept–property pair as follows (note that the counts used to populate the table cells are based on the number of *distinct* tuples containing concepts and properties, not on total corpus token counts):

$$\mathrm{llr} = 2\sum_{ij} O_{ij} \log \frac{O_{ij}}{E_{ij}}$$

The log-likelihood ratio approximates a χ^2 distribution with one degree of freedom (Evert, 2005). Based on this approximation, in Strudel we keep only those pairs with a log-likelihood ratio above 19.51 (corresponding to a probability below 0.00001 under the assumption of independence) and where $O_{11} > E_{11}$ (i.e., association is significantly above, not below chance).

We chose the log-likelihood ratio as a concept–property association measure because it is widely used in computational linguistics and it has been shown to be robust against data sparseness, providing reasonable significance-of-association estimates also in the presence of low counts in the contingency table cells (Dunning, 1993).

3.2. Property typing with type sketches

In the next step, we provide a pattern-based characterization of the relation occurring between a concept and a property by generalizing across similar patterns that connect them, and keeping track of the distribution of these generalized patterns in what we call the type sketch of the pair (because the generalized patterns are seen as surface cues to property types). We are following here a long tradition in lexical semantics proposing that semantic relations can be captured directly by the explicit syntactic material expressing them (see, most notably, Levi, 1978). We store the whole type distribution because this is useful for disambiguation purposes (*in* might cue hypernymy in a sketch with *such as*, but location if it occurs with *inside*).

Generalization is performed by another simple rule-based module (presented in full in the technical report, and amounting to 12 rules implemented as regular expression-based substitutions) that looks for prepositions, verbs, and other ''meaningful'' components of a pattern and discards the rest. Consider a hypothetical concept–property pair occurring with the following patterns (concept always on the left): *with a number of* (two times), *with a* (one time), *with ADJ* (one time), *have* (one time), and *has* (one time). The type sketch for this pair would be: *C_with_P* (66.6%), *C_have_P* (33.3%). Illustrative examples of the Strudel output, including type sketches, are presented in Table 2, where properties are annotated with POS; *log-likelihood* is the concept–property association score; types also record the

Table 2
Examples of Strudel output with type sketches

Concept	Property	Log-likelihood	Type Sketch
child	parent-n	11,726.7	P_of_C (40%), P_with_C (11%)
child	parent-v	120.8	P_C (79%)
lion	mane-n	259.1	C_'s_P (50%), C_with_P (15%), C_have_P (12%), P_of_C (10%)
wolf	forest-n	78.3	C_in_P (32%), P_of_C (31%), C_through_P (14%)
wolf	pack-n	251.2	P_of_C (70%), C_in_P (15%)
egg	female-n	1,603.4	P_produce_C (13%), C_by_P (12%)
breakfast	croissant-n	257.2	P_for_C (46%), C_of_P (34%), C_with_P (12%)
beach	walk-v	687.6	P_C (29%), P_from_C (24%), P_along_C (23%), P_on_C (13%)
grass	green-a	277.6	P_C (58%), C_is_P (25%), C_is_ADV_P (16%)

relative ordering of the concept and property, and only types accounting for at least 10% of the distribution are reported.

4. Model training

All models were trained on a lemmatized and POS-tagged version of ukWaC (Baroni, Bernardini, Ferraresi, & Zanchetta, 2009), a very large corpus of English (almost two billion tokens) collected via a random-like crawl of the uk domains of the Web. Statistics were collected for a list of 1,234 concrete concepts (or, more precisely, lemmas associated with concrete concepts). This set was created by merging lists of concepts found in the experimental literature and all nouns in a corpus of child-directed speech (Rowland, Pine, Lieven, & Theakston, 2005) that belonged, in all their senses, to a concrete superordinate category according to WordNet (Fellbaum, 1998).

Strudel does not require much parameter tuning: most of the often arbitrary work done by window size or dimension selection in other systems is implicitly taken care of by the pattern grammar (see the Supporting Information for details).

For the SVD model, we collected co-occurrence statistics among content words only (Rapp, 2003, 2004, to a similar effect, lemmatizes the corpus and filters out function words through a stop list). Co-occurrences were collected within a window of 20 words, using the 5,000 most frequent words as contexts, and raw occurrence counts as input to the SVD. Other settings of these parameters were also tried, obtaining inferior results.

The dimensionality of the term-by-term co-occurrence matrix was reduced by SVD (see, e.g., Manning & Schütze, 1999, pp. 558–564; Schütze, 1997, pp. 189–195). As this operation benefits from patterns of correlation among columns (contexts) in terms of rows (target items), we used a longer list of rows than for the other models. This included the 1,234 concepts of interest plus all content words that occur at least 1,000 times in ukWaC (with the exception of the top 10 most frequent words). The resulting set contained 22,682 words. After SVD, we picked as our dimensions the top 300 left singular vectors (multiplied by the corresponding singular values). Preliminary experiments suggested that picking more or fewer vectors had little impact on performance. Landauer and Dumais (1997) show that 300 singular vectors are an optimal choice, at least on the TOEFL task, and Rapp also used 300 singular vectors.

The dependency vectors (DV) model was prepared by dependency-parsing ukWaC with MINIPAR[1] and feeding it to the DV toolkit.[2] We used the parameter settings that were chosen in Padó and Lapata (2007) as most effective: *medium* context, *length* path function, and *log-likelihood* weighting. We experimented with context vectors of various dimensions, consistently obtaining the best results with 10,000 dimensions.

The attribute value (AV) model was trained with the attribute- and value-harvesting patterns of Almuhareb and Poesio (2004): *the ATTRIBUTE of the CONCEPT [is|was]* (e.g., *the speed of the car is*) and *[a|an|the] VALUE CONCEPT [is|was]* (*a fast car is*). Following Almuhareb and Poesio, we used *t*-scores to measure the association between concepts and attributes/values. We selected as dimensions the 5,365 harvested items (attributes or values)

that had *t*-scores above 1.28 ($p < .1$) with at least one target concept. Each concept was represented in this space by a vector of *t*-scores. In informal experimentation, we found that raising the 1.28 threshold (thus being more selective) hampered performance; lowering it did not have a large impact on the tasks.

5. Concept description examples

In order to get an intuitive feeling of how Strudel and other CSMs ''describe'' concepts through their most salient properties, we report here the top 10 properties (according to the respective scores) produced by each model for three example concepts, namely *book* (Table 3), *motorcycle* (Table 4), and *tiger* (Table 5). The examples were chosen before looking at the model outputs. For comparison, we also report the 10 properties produced by the largest number of informants in the elicited norms collected by McRae and colleagues (see Section 6). For Strudel, we list up to four generalized types (and, in any case, only types accounting for at least 10% of the distribution). For SVD, the ''properties'' are the nearest neighbors in the reduced dimensionality space according to the cosine measure.

Strudel provides a very plausible description of books (reasonable properties with reasonable type sketches): books *are written*, *published*, and *read*, they are *by an author*, *from a publisher*, *for a reader*, and *on a subject*, they *have pages* and *chapters*, and they are *in libraries*. Strudel is the only CSM that discovered that books are related to libraries, chapters, publishers, pages, and subjects. The AV and DV models found rather generic terms that in the case of DV also include prepositions. SVD provides properties that are largely complementary to the ones found by Strudel (such as *article*, *excerpt*, and *commentary*).

Concerning *motorcycles*, results are not as clean. The performance of Strudel is comparable with the one of SVD and DV, and we clearly see here the tendency of Strudel, when compared with the human judgments in the norms, to prefer actional and situational properties (*riding*, *parking*, *colliding*, being *on the road*) over parts (such as *wheels* and *engines*). Properties expressed by verbs tend to be highly ranked by Strudel as verbs will occur in a variety of different contexts around the target concept (with or without auxiliary, in different tenses, with or without adverbs, adjectives or determiners in between, etc.), and Strudel, as discussed in Section 3.1, is sensitive to context variety, rather than absolute frequency of co-occurrence. We will come back to this distinction between the norms and Strudel in Section 6.

As far as *tigers* are concerned, Strudel is the only CSM capturing their most salient visual feature (*having stripes*), as well as their saliently ferocious behavior (*killing* and *mauling*) and typical locations (*jungles*, *zoos*, and *cages*). The main limitation of SVD-like models as property-based descriptions is clearly illustrated by this example: nearest neighbors are not properties but rather concepts that share properties with the target, and thus, mostly, taxonomic coordinates: *gorillas*, *elephants*, etc. (Baroni & Lenci, 2008).

Although in our tests below DV will emerge as the best performing model after Strudel, a qualitative look at the properties it generates suggests that they often are

Table 3
Properties of *books*

Model	Property	Type Information
McRae	pages	has
	reading	used by
	words	has X in it
	authors	has
	libraries	found in
	hard cover	has a
	paper	made of
	tells stories	inanimate behavior
	schools	found in
	soft cover	has a
Strudel	reader-n	C_for_P (30%), P_of_C (11%)
	author-n	P_of_C (70%), C_by_P (12%)
	read-v	P_C (79%), C_P (15%)
	library-n	P_of_C (25%), C_in_P (23%), C_from_P (16%)
	chapter-n	P_of_C (35%), P_in_C (26%)
	write-v	P_C (63%), C_P (31%)
	publish-v	C_P (53%), P_C (40%)
	publisher-n	P_of_C (35%), C_from_P (17%)
	page-n	P_of_C (43%)
	subject-n	C_on_P (71%), P_of_C (11%)
AV	whole-a	value
	first-a	value
	title-n	attribute
	entire-a	value
	bulk-n	attribute
	aim-n	attribute
	focus-n	attribute
	log-n	value
	order-n	value
	rest-n	attribute

Model	Property	Model	Property
SVD	reader-n	DV	read
	reading-n		of
	story-n		write
	article-n		publish
	writing-n		his
	excerpt-n		book
	writer-n		review
	commentary-n		in
	write-v		buy
	interesting-a		journal

generic collocates of the target word. For *tigers*, these collocates pertain in part to metaphorical usages of the word. Curiously, for both *motorcycle* and *tiger* the top DV property is the concept itself.

Table 4
Properties of *motorcycles*

Model	Property	Type Information
McRae	wheels	has
	2wheels	has
	dangerous	is
	engine	has an
	fast	is
	helmets	used with
	Harley Davidson	made by
	loud	is
	1 or 2 people	used by
	vehicle	a
Strudel	ride-v	P_C (80%), C_P (16%)
	rider-n	P_of_C (34%)
	sidecar-n	C_with_P (67%), C_without_P (16%), P_of_C (11%)
	park-v	C_P (81%), P_C (19%)
	road-n	C_on_P (51%)
	helmet-n	C_without_P (33%), P_on_C (33%), P_if_C (11%), P_than_C (11%)
	collision-n	C_in_P (70%), P_with_C (12%)
	vehicle-n	such_P_as_C (19%)
	car-n	C_with_P (21%), C_as_P (13%)
	moped-n	P_for_C (62%), C_as_P (12%), C_use_P (12%), P_in_C (12%)
AV	first-a	value
	replacement-n	value
	large-a	value
	provisional-a	value
	classic-a	value
	damaged-a	value
	driver-n	attribute
	diesel-a	value
	traditional-a	value
	structure-n	attribute

Model	Property	Model	Property
SVD	motor-n	DV	motorcycle
	wheeler-n		accident
	driver-n		car
	scooter-n		insurance
	moped-n		scooter
	motorbike-n		ride
	driving-n		rid
	motor-v		racing
	vehicle-n		motor
	racetrack-n		helmet

Table 5
Properties of *tigers*

Model	Property	Type Information
McRae	stripes	has
	carnivore	a
	teeth	has
	animal	an
	jungles	lives in
	feline	a
	Africa	lives in
	dangerous	is
	black	is
	circuses	used in
Strudel	jungle-n	C_in_P (53%), C_through_P (11%)
	zoo-n	C_in_P (60%), C_from_P (10%)
	lion-n	P_on_C (46%), C_because_P (15%)
	maul-v	P_by_C (47%), C_P (47%)
	kill-v	C_P (51%), P_C (25%), P_by_C (19%)
	stripe-n	P_of_C (23%), P_on_C (23%)
		C_have_P (15%), C_with_P (15%)
	cage-n	C_in_P (55%), C_'s_P (14%)
	specie-n	such_P_as_C (47%), P_of_C (17%), P_like_C (10%)
	habitat-n	P_of_C (47%), C_'s_P (26%), P_for_C (13%)
	extinction-n	C_from_P (58%), P_of_C (17%), C_towards_P (17%)
AV	white-a	value
	Siberian-a	value
	Celtic-a	value
	male-a	value
	brave-a	value
	Bengal-n	value
	character-n	attribute
	wounded-a	value
	Indian-a	value
	Chinese-a	value

Model	Property	Model	Property
SVD	elephant-n	DV	tiger
	gorilla-n		lion
	lion-n		cub
	kangaroo-n		shark
	snake-n		elephant
	crocodile-n		cooperative
	alligator-n		economy
	hunter-n		oscar
	hunt-v		white
	monkey-n		portfolio

Finally, the value properties uncovered by AV are often generic (*first motorcycle*, *male tiger*), and this is probably the model producing the least convincing overall descriptions. However, if we focus specifically on the AV attributes we find, intriguingly, very abstract characterizations missed by all other models, such as the *title*, *aim*, and *focus* of *books* and the *structure* of *motorcycles*.

6. Property generation

Having verified in the previous section that Strudel produces sensible property-based concept descriptions from a qualitative point of view, we turn now to a quantitative evaluation of the properties generated by the algorithms. We compare the highest scored properties produced by our CSMs with the properties produced by the largest number of subjects in the norms compiled by McRae, Cree, Seidenberg, and McNorgan (2005) by asking informants to generate a list of up to 10 defining or typical properties of various concepts.[3] We are particularly interested in this sort of comparison as such subject-generated property norms (see also Garrard, Lambon Ralph, Hodges, & Patterson, 2001; Vinson & Vigliocco, 2008) have been extensively used in the psychological literature as proxies for mental concept descriptions in experimental and simulation-based work (similar comparisons are carried out by Almuhareb, 2006; Poesio et al., 2007; see Schulte im Walde, 2008, for comparisons of CSMs with associative norms).

In the task presented in this section, the properties in the McRae norms are used as the gold standard that models are compared against. The term ''gold standard,'' in this context, must be taken with a grain of salt. The norms collected by McRae and colleagues are the product of a concept description elicitation experiment during which subjects presumably access their conceptual knowledge, but (as their collectors readily recognize) they are certainly not direct windows into the ''true'' representation of concepts in the mind of speakers. Still, by looking at the examples in the tables of Section 5, the concept descriptions in the norms look intuitively plausible (more so than those produced by any CSM) and, more importantly, the extensive literature using speaker-generated properties in simulations and experiments (see McRae et al., 2005, for references) has proved their usefulness as surrogates of mental representations of concepts. Thus, we see our attempt to approximate subject-generated properties with CSMs as a reasonable first step towards the production of corpus-based concept descriptions. At the same time, we will not look at systematic mismatches between subject- and corpus-generated properties as errors in the latter, but as potentially informative cues of how the two property harvesting methods differ.

6.1. Experimental setting

We studied the properties of 44 concrete concepts from the McRae property norms that are available in machine-friendly format from http://wordspace.collocations.de/doku.php/esslli:start. The target concepts are as follows: (animals) *chicken*, *eagle*, *duck*, *swan*, *owl*, *penguin*, *peacock*, *dog*, *elephant*, *cow*, *cat*, *lion*, *pig*, *snail*, *turtle* (fruit and vegetables)

cherry, *banana*, *pear*, *pineapple*, *mushroom*, *corn*, *lettuce*, *potato*, *onion* (everyday objects) *bottle*, *pencil*, *pen*, *cup*, *bowl*, *scissors*, *kettle*, *knife*, *screwdriver*, *hammer*, *spoon*, *chisel*, *telephone* (vehicles) *boat*, *car*, *ship*, *truck*, *rocket*, *motorcycle*, *helicopter*. For each target concept, we picked the top 10 properties (ranked by number of subjects that produced them). We limited ourselves to the top 10 human-generated properties of each concept as for about 10% of the target concepts the norms only contain 10 properties (for *snails*, they list nine properties). Given the top 10 ranked properties generated by a model, we computed precision for each concept with respect to the norms-based gold standard, and we averaged precision across the 44 concepts (precision here is simply the proportion of top 10 model-ranked concepts that are also in the gold standard).

Application of the AV, DV, and Strudel models to the task is straightforward—we just pick the top n properties, as ranked by t-score in the first case, and by log-likelihood ratio in the other two. For SVD, we pick as candidate properties the nearest neighbors of a concept in the Euclidean space defined by the reduced dimensions (again, using the cosine measure).

6.2. Results

Table 6 reports percentage average precisions and standard deviations for the 10-best lists generated by each model matched against the McRae norms-based gold standard.

The advantage of Strudel over its nearest competitor (DV) in terms of precision across the target concepts is highly significant (paired t-test, $t = 5.2$, d.f. $= 43$, $p < .0001$). Guessing on average 23.9% of the most salient properties produced by humans should be seen as a rather remarkable result, given the completely unconstrained nature of the task. Indeed, in experiments in which we used simple property-matching heuristics to maximize overlap, we found that there was never more than a 30% pairwise overlap between the properties of shared concepts found in human-generated norms collected by different researchers (Poesio et al., 2007).

The mismatches between the norms and Strudel are rather systematic, and they are not necessarily Strudel errors. As we already saw in the examples of Section 5 and we will see again in the analysis of the categorization experiment (in Section 8), Strudel has a strong tendency to favor activity-related properties. The norms instead contain more parts and qualities. As a fairly typical example, for the concept of *car*, both Strudel and the norms list *engine*, *gasoline*, and *transportation* among the top properties. Only the norms feature *(4) wheels*, *doors*, *steering wheel*, *expensive*, *passengers*, and *vehicle*, whereas only Strudel produces *driven*, *driver*, *parked*, *road*, *garage*, *race*, *parking*. This confirms the observation by Baroni and Lenci (2008) that distributional models tend to underestimate part and surface

Table 6
Precision of 10-best property lists against the McRae norms

Model	Avg. Precision	*SD*
Strudel	23.9	11.3
DV	14.1	10.3
AV	8.8	9.9
SVD	4.1	6.1

properties with respect to subject-elicited concept descriptions (an example of the first type would be *wheels* for the *car* concept, whereas *being yellow* is a surface property of *bananas*).

Parts and surfaces are highly visually accessible, and thus it makes ecological sense that evidence coming from verbal input, as encoded in a CSM, would rather pertain to other types of properties (such as actions, functions, superordinates) that are not perceptual, or that require some higher level structuring of the scene being perceived. We also cannot exclude that the perceptual features are emphasized in the human norms because of the instructions given to subjects (McRae et al., 2005 report that subjects were asked to list physical/perceptual and functional properties, categories, and encyclopedic facts).

7. Discovering property types

We have informally argued that type sketches cue the semantic relation between a concept and a property. By looking at the examples in Section 5, the types make intuitive sense. However, most connectors are very general and ambiguous expressions, and their plausibility derives from the interpretation that we, as readers of the example tables, are inclined to provide. The preposition *in* in the type sketch of *tiger–zoo* is interpreted as a cue of a locative relation, whereas *in* as a type of *tiger–animal* is naturally seen as a marker of class membership. Do the type sketches (note that they contain a whole *distribution* of types) partition concept–property pairs into groups that correspond, roughly, to coherent property types, or are we just using our own intuitions to give a meaning to the types? To investigate this issue, we exploited the fact that the norms of McRae and colleagues have been manually annotated by the experimenters with property types such as category, function, and part. We can thus perform unsupervised clustering of a set of concept–property pairs produced by Strudel and check the clustering results against the manual property type classification from the norms. Note that here we are using as our reference set not the properties that were produced by the McRae subjects, but the labels that were assigned by McRae and colleagues to the concept–property pairs generated by the subjects, that is, a categorization produced by expert annotators that fully qualifies as the ‘‘gold standard.’’

The results will show that, while not perfect, type sketch clustering achieves a very promising performance level in reconstructing the manual labels. In turn, this suggests that type sketches are indeed good indicators of the semantic relation linking a concept and a property, that is, Strudel is not only extracting properties, but it is also implicitly *typing* those properties in terms of the semantic relation they hold with the concept. This is an important first step from flat lists towards producing more structured concept descriptions, of the sort that have been widely adopted in cognitive science (Murphy, 2002, Chapter 4).

7.1. Experimental setting

We extracted from the McRae norms subset described in Section 6.1 those concept–property pairs that were also found by Strudel. As the Strudel verbal, adjectival, and nominal

properties are currently characterized by nonoverlapping types, and for nominal properties we obtained a larger number of pairs, organized into a more granular typology, we decided to focus on nouns, and in particular on the following four property types (the ones for which we could extract more than 20 representative pairs): part (*motorcycle–engine*), category (*pineapple–fruit*), location (*pig–farm*), and function as expressed by nouns (*ship–cargo*). The resulting data set consists of 205 concept–property pairs containing 44 distinct concepts (Section 6.1) and 123 distinct properties. According to the norms, the properties in the pairs include 85 parts, 31 categories, 25 locations, and 64 functions.

For each concept–property pair in the data set, we build the vector representing its type sketch (i.e., how many times the pair occurs with each generalized type), leading to a matrix of 205 concept–property pairs by 215 generalized types. A few example columns from this matrix, for the rows corresponding to the concept–property pairs *cup–drinking*, *boat-fishing*, *swan–lake*, and *boat-ocean*, are reported in Table 7. The full matrix, with row and column labels and the clustering results, is available as part of the Supporting Information.

The pairs are clustered in the generalized type space using the CLUTO toolkit,[4] a widely used package that implements various clustering and clustering analysis techniques. In particular, here and below, we perform clustering using one of the most basic partitional clustering techniques, that is, the *k*-means algorithm that starts by assigning the data points to *k* random clusters, then iteratively re-assigns the points to the nearest cluster until convergence (Manning & Schütze, 1999, Chapter 14). We accept all default parameters of CLUTO for the *k*-means option (Karypis, 2003). Here and below, similarity between vectors is measured by the cosine of the angle they form (see, for example, Manning & Schütze, 1999, Section 8.5.1).

We measure clustering quality by looking at the cluster-by-true class confusion matrix, and quantitatively in terms of percentage *purity* (Zhao & Karypis, 2001). If n_r^i is the number of items from the *i*th true (gold standard) class that were assigned to the *r*th cluster, *n* is the total number of items and *k* the number of clusters, then:

$$\text{Purity} = \frac{1}{n}\sum_{r=1}^{k}\max_i(n_r^i)$$

Expressed in words, for each cluster we count the number of items that belong to the true class that is most represented in the cluster (the ‘‘majority’’ class), and then we sum these

Table 7
Example rows and columns from the concept–property by generalized-type matrix

	Column Labels					
Row Labels	C_during_P	P_from_C	C_in_P	P_of_C	C_on_P	P_with_C
cup-drinking	1	16	0	5	0	0
boat-fishing	0	77	2	2	0	10
swan-lake	0	0	5	1	11	4
boat-ocean	0	0	3	0	9	0

counts across clusters. The resulting sum is divided by the total number of items so that, in the best case (perfect clusters), purity will be 1 (in percentage terms, 100%). As cluster quality deteriorates, purity approaches 0.

With a clustering solution, CLUTO also produces a list of the most ''descriptive'' features of the clusters it generates, determined by selecting the dimensions of the clustered vectors that contribute the most to the average similarity between the objects of each cluster (Karypis, 2003). We will look at these CLUTO-generated features in our qualitative analysis of the results. As the features in the current analysis are generalized types from the Strudel type sketches, we will refer to them as descriptive types. In this experiment, the only appropriate algorithm is Strudel (DV and SVD do not have types; AV only distinguishes two types).

7.2. Results

The purity of clustering into four classes is at a rather high 80%, that is, on average 80% of the items in a cluster come from the same true class (the expected purity in case of random assignment of our test items to four clusters is just 41%). The confusion matrix of the clustering solution is reported in Table 8, together with the top five most descriptive types of each cluster.

From the distribution of true classes across clusters and the descriptive types, we see that Strudel has done a very good job to discover, in a completely unsupervised way, part, category, and location types. Function is more problematic, with half of the gold standard functional relations outside the ''function'' cluster (cluster 4). Some of these mismatches correspond to reasonable alternative classifications of the relations (*mushroom–hallucinogen* in the category cluster, *ship–cargo* in the part cluster, *cherry–cake* in the location cluster), but others are not (*lettuce–salad* as part, *knife–cutting* as location, etc.). On closer inspection, function as expressed by single noun properties in the gold standard is a mixed bag, including activities (*pen–writing*), objects and participants involved in the functional activity (*pen–page*, *pen–author*), and fillers of prototypical functions (*cup–tea*). This dishomogeneity might explain at least in part the difficulties of Strudel with the function type.

8. Categorization

Having shown, qualitatively and quantitatively, that Strudel produces plausible property-based descriptions of concepts, and moreover that its typing mechanism is grouping

Table 8
Confusion matrix of property clustering with most descriptive types of each cluster

Cluster	Part	Category	Location	Function	Descriptive Types
1	**77**	0	1	10	P_of_C, C_'s_P, C_with_P, C_have_P, P_on_C
2	1	**31**	0	1	P_such_as_C, P_like_C, C_as_P, P_as_C, P_from_C
3	2	0	**24**	21	C_on_P, P_with_C, C_in_P, C_from_P, C_to_P
4	5	0	0	**32**	C_of_P, P_from_C, C_for_P, P_in_C, P_by_C

properties into coherent property types, we now move from infra-conceptual to inter-conceptual relations, and we tackle a traditional CSM task in which concept descriptions are used to measure cross-concept similarity in order to categorize basic-level concepts into superordinates. Categorization, the ability to group similar entities into higher level classes, is probably the core skill in semantic cognition, as it allows us to generalize, see commonalities, and draw inferences. Categorization is, therefore, an important first step towards simulating more complex cognitive skills.

8.1. Experimental setting

We constructed a test set of 10 common concrete categories extracted from the norms of Van Overschelde, Rawson, and Dunlosky (2004). For each superordinate category, we selected up to 10 concepts, as ordered by typicality rating according to the norms, and that were also attested in our concept list and in the McRae norms. The resulting test set contains the following 83 concepts:

(land) mammals: dog, elephant, cat, cow, lion, pig, horse, bear, tiger, deer
birds: robin, sparrow, pigeon, chicken, eagle, duck, owl, penguin, hawk, crow
fish: cod, goldfish, minnow, salmon, trout, tuna
vegetables: broccoli, spinach, lettuce, potato, onion, carrot, cucumber, peas, celery, beans
fruit: apple, orange, grape, peach, strawberry, plum, grapefruit
trees: oak, pine, birch, cedar
vehicles: boat, car, ship, truck, motorcycle, helicopter, bus, aeroplane, train, bicycle
clothes: shirt, pants, socks, jacket, sweater, bra, skirt, coat, dress, blouse
tools: screwdriver, hammer, chisel, wrench, sandpaper, pliers
kitchenware: cup, bowl, spoon, pan, pot, blender, ladle, strainer, colander

Concepts were represented by vectors whose dimensions are properties (for SVD: latent dimensions), filled by the relevant scores (e.g., the Strudel vectors contain log-likelihood ratios).

To build the Strudel vectors, we used typed properties as dimensions, by prefixing properties with the generalized types that account for the largest portion of the distribution in the corresponding type sketches. For example, as the most common type in the type sketch of *reader* as a property of *books* is *for* (see Table 3), the Strudel vector representing the *book* concept would have a dimension corresponding to *for reader*. The intuition behind this approach is that the same property coupled with different types express different relations that characterize different concepts (*for reader* is a property of *books*, but *by reader* might be a property, e.g., of *reviews*). Examples of columns of the resulting 83×7,953 matrix for the rows corresponding to the *car*, *truck*, *dog*, and *cat* concepts are presented in Table 9 (using plain English translations of the column labels). The full Strudel matrix (with row and column labels, and the clustering results) is available in the Supporting Information. We leave it to future research to explore other ways to build vectors out of properties and type sketches that exploit the full distribution of types in the sketches.

Table 9
Example rows and columns from the concept by typed-property matrix

	Column Labels					
Row Labels	C is trained	C at speed	C on leash	C on street	C is euthanized	C's windshield
car	0	738.0	0	326.9	0	32.9
truck	0	41.5	0	0	0	0
dog	438.6	0	279.8	0	41.6	0
cat	0	0	0	0	58.7	0

The norms were treated here as a human-based computational semantic model, where concepts are represented by vectors with speaker-generated properties as dimensions and the number of subjects that produced each property as values. We refer to this model with the uppercase label NORMS, to underline the fact that it is a computational simulation derived from the McRae norms, as opposed to categorization data directly elicited from humans.

As in Section 7.1, we performed *k*-means clustering of the concept vectors with the default CLUTO parameters and used percentage purity as our quantitative evaluation measure. As we are also interested in how the models group classes hierarchically, we exploited CLUTO's option to perform agglomerative clustering on top of the partitional solution. The agglomerative solution was built by combining, at each step, the two clusters that have, on average, the highest pairwise similarities among their members.

When interpreting the results, we must keep in mind that CLUTO provides ''hard clustering'' solutions that are good for a clear-cut quantitative evaluation but do not permit multiple cluster membership. Further work should explore soft clustering methods that might reveal a more nuanced picture than the one emerging from the current results. For example, we will see that NORMS treated *chicken* as an animal, whereas Strudel placed it in the ''food'' category. Both classifications are reasonable, and a soft clustering technique might reveal that for both models *chicken*, being polysemous, is actually halfway between animals and food.

8.2. Results

Table 10 reports the dimensionality of the vectors used for clustering with each model (the 300 SVD dimensions result from reducing a 22,682×5,000 matrix) and clustering performance in terms of percentage purity.

Table 10
Concept clustering results

Model	Dimensions	Purity
NORMS	603	97
Strudel	7,953	91
DV	10,000	79
SVD	300	71
AV	1,249	45

NORMS outperforms all corpus-based models with nearly perfect clustering. Strudel emerges as the leading CSM, with purity above 90%, more than 10% ahead of the next best CSM, namely DV.

Table 11 reports the cluster-by-true category confusion matrix for the Strudel clustering solution, together with the full list of ''outliers,'' that is, concepts that do not belong to the majority category of the cluster they were assigned to.

Spoons are confused with tools (arguably, they have more in common with, say, *screwdrivers* than *bowls*, although the same could be said for *ladles*, that are clustered with the rest of the kitchenware). *Horses* are clustered with vehicles, which is not unreasonable: Strudel, as we said, has a tendency to highlight activity-related properties, that in turn often stress the functional aspects of concepts (what you do with them). Along similar lines, *chickens*, the only commonly eaten bird in our test set, end up with vegetables—we can interpret this as a case of unintended polysemy, where Strudel picked the food sense of *chicken* (that is probably much more common in Web texts), and formed a ''salty food'' cluster including *chicken* and vegetables. With *penguins* (and *ducks*) in the land mammal cluster, the bird cluster might be more appropriately labeled as a ''flying things'' group, also including *aeroplanes* and *helicopters* (*aeroplanes* are in the bird cluster in the NORMS solution as well).

We proceed now to an analysis of the properties of the superordinates discovered by Strudel and other models, and then of how the superordinates cluster into higher level categories. We stress that from now on, when we refer to categories, we mean the clusters that have been induced by the models from the data, labeled by their majority classes. For example, when we refer below to the Strudel bird superordinate, this is not our manual set of birds, but cluster 2 of Table 11, including *aeroplanes* and *helicopters* and missing *ducks*, *penguins*, and *chickens*.

8.2.1. Properties of superordinates

Table 12 reports the top five most descriptive properties (i.e., the most descriptive features returned by CLUTO as part of the clustering procedure, as explained in Section 7.1) of

Table 11
Confusion matrix of Strudel concept clustering and full list of items not in majority class of each cluster

Clust	Mam	Bird	Fish	Veg	Fruit	Tree	Vehic	Cloth	Tool	k-ware	Outliers
1	**9**	2	0	0	0	0	0	0	0	0	duck, penguin
2	0	**7**	0	0	0	0	2	0	0	0	aeroplane, helicopter
3	0	0	**6**	0	0	0	0	0	0	0	
4	0	1	0	**10**	0	0	0	0	0	0	chicken
5	0	0	0	0	**7**	0	0	0	0	0	
6	0	0	0	0	0	**4**	0	0	0	0	
7	1	0	0	0	0	0	**8**	0	0	0	horse
8	0	0	0	0	0	0	0	**10**	0	0	
9	0	0	0	0	0	0	0	0	**6**	1	spoon
10	0	0	0	0	0	0	0	0	0	**8**	

Table 12
Most descriptive properties of induced superordinates

Superordinate	Model	Descriptive Properties
mammal	NORMS	animal, large, has 4 legs, has fur, has legs
	Strudel	in zoo, seen, killed, shot, breeds
	DV	dog, cat, chicken, sheep, duck
bird	NORMS	bird, flies, has feathers, has wings, has beak
	Strudel	flies, bird, nests, has nest, in sky
	DV	tiger, eagle, lion, hawk, owl
fish	NORMS	fish, swims, in water, edible, has gills
	Strudel	catching, in river, in pond, fish, in shoal
	DV	salmon, park, trout, robin, color
vegetables	NORMS	vegetable, green, in salads, edible, nutritious
	Strudel	cooked, eaten, chopped, sliced, vegetable
	DV	carrot, tomato, onion, bean, cabbage
fruit	NORMS	fruit, juicy, sweet, on trees, red
	Strudel	eaten, fruit, has juice, has flavor, has aroma
	DV	orange, apple, tree, cherry, juice
tree	NORMS	tree, has leaves, in forests, tall, for making furniture
	Strudel	in forest, has wood, grows, planting, in woodland
	DV	tree, oak, wood, pine, house
vehicle	NORMS	for transport, has wheels, for passengers, has engine, large
	Strudel	ridden, passenger on it, on road, has driver, parks
	DV	fly, on, by, ride, station
clothes	NORMS	clothing, worn by women, various colors, has sleeves, for warmth
	Strudel	worn, knitted, with collar, with sleeve, dressing in it
	DV	wear, trouser, white, shirt, skirt
tool	NORMS	tool, made of metal, has handle, has head, in toolboxes
	Strudel	used, to screw, needs socket, has bevel, tool
	DV	with, hammer, fine, wooden, jaw
kitchenware	NORMS	in kitchens, made of metal, of plastic, for cooking, has handle
	Strudel	ingredient in it, has water, oil in it, plant in it, drunk
	DV	in, into, bowl, pan, pot

each category discovered by the NORMS, Strudel, and DV models. The descriptive features produced by CLUTO for the SVD model are uninterpretable latent dimensions, and the AV's output is difficult to interpret due to poor clustering quality. Those in Table 12 can be seen as salient properties of the emergent superordinates that the models discovered in an unsupervised manner, and thus they represent a step forward from classic categorization results in corpus-based semantics. We are not just discovering superordinate concepts, but we are also describing them in terms of characteristic properties. Note that in the table we converted the NORMS and Strudel types to plain English phrases.

Both NORMS and Strudel provide mostly reasonable properties for the superordinate concepts, and we confirm the tendency of Strudel, that we already observed in Sections 5 and 6 above, to produce ''actional'' and ''situated'' descriptions, whereas the NORMS produce more parts, qualities, and categories. For example, the superordinate vegetable concept

is described by NORMS as *green*, found *in salads*, *edible*, and *nutritious*, whereas Strudel captures it largely through our interaction with it: it is *cooked*, *eaten*, *chopped*, and *sliced* (this characterization makes sense also in light of the fact that *chicken* is for Strudel in the ''vegetable'' class); the most distinctive Strudel characteristic of vegetables as opposed to fruit is that the former is *cooked*. The difference between NORMS and Strudel is most striking for mammals. For NORMS, a mammal is a *large animal with four legs* and *fur*, for Strudel it is an entity that is found *in zoos*, it is *seen*, *killed*, *shot*, and it *breeds*. These two models are producing largely complementary descriptions, and the DV model is providing yet another style of description, but one that, at least on an intuitive level, is less convincing. Essentially, classes are described in the latter by lists of hyponyms, related entities and, in some cases, prepositions.

Like for basic concepts, we can use Strudel type sketches to characterize the relation between a superordinate concept and its properties. Putting together the results on type sketch clustering of Section 7.1 with the idea of superordinate type sketches, we can now show that Strudel is not only capable of assigning meaningful properties to the superordinates it discovers but also to provide an indication of the type of properties that characterize the superordinates. We proceed as follows. From the CLUTO output, we pick the top 20 most descriptive properties for each cluster (superordinate) of the concept categorization experiment. For each of these, we compute the type sketch of the property in relation to the superordinate by averaging the weights across the type sketches of the property in combination with the subordinate concepts. We then measure the similarity of the resulting superordinate type sketches to the average type sketches of the four property type clusters extracted as described in Section 7.1 (nominally expressed parts, categories, locations, and functions;

Table 13
Properties of Strudel superordinates strongly associated with a property type

Superordinate	Type	Descriptive Properties
mammal	category	animal
	location	pond, farm
bird	category	bird
fish	category	fish
vegetables	category	vegetable
fruit	part	skin, hint
	category	fruit
tree	part	trunk, bark
	category	tree
	location	oak
vehicle	part	crew, wheel
	location	road
clothes	part	pocket, sleeve
tool	part	jaw, bevel
	category	tool
	location	anvil, screw
kitchenware	function	tea, soup, coffee, rice, water

again, we work with the clustering-determined classes, not the gold standard ones). Table 13 shows those descriptive properties (among the top 20 of each cluster) that, when connected to a superordinate, have a cosine of at least 0.7 (an arbitrary threshold) with one of these four prototype vectors. We want to stress again how, in this analysis, the superordinates, their descriptive properties, the property type prototypes, and the association of properties to prototypes were all discovered in a completely unsupervised fashion.

Looking at Table 13, we can first of all observe that types are assigned with very high precision: of 28 properties, only three are plainly wrong in terms of property type (*hint* as a part of fruit, *oak* as location of trees, and *screw* as the location of tools), and all the correct ones seem reasonably prototypical for the corresponding concepts. Recall is low, but it could be raised using longer lists of top descriptive properties and/or a lower similarity threshold (at the cost of lower precision). Strudel is particularly good at category naming, producing a category name for seven of the 10 superordinates, always correct. Mammals are labeled with the super-superordinate *animal*. This is appropriate because the empirical cluster contains two non mammals (*ducks* and *penguins*), and more in general *animal* is an intuitively more common label than the slightly technical *mammal*. Interestingly, Rogers and McClelland (2004, chapter 5), in their naming simulations, postulate an intermediate category name for birds and fish, but for mammals they only assume the general *animal* category label. Moreover, for our 10 (real) mammals, the McRae norms record *mammal* as one of the top 10 subject-generated properties in one case only (*cows*), whereas *animal* is among the top properties for six of them.

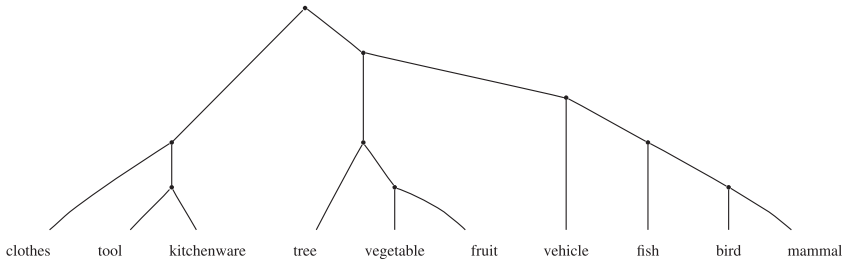
Of the three missing names, clothes have *dress* as the closest property to the category prototype vector, but with a cosine of just about 0.2. This leaves out only vehicles and kitchenware, and for the latter category there is arguably no natural single-word superordinate name (Strudel misses composite properties such as *kitchen item* by design).

For parts, it is interesting to observe how, despite the fact that Strudel is in general quite bad at collecting them in undifferentiated property lists, by applying type-based filtering we were able to harvest quite a few parts: vehicles *have wheels*, clothes *have pockets*, etc. This is another example of how, thanks to type sketches, Strudel can be tuned to different tasks (in this case, looking for parts) without the need to go back to the corpus to retrain the model.

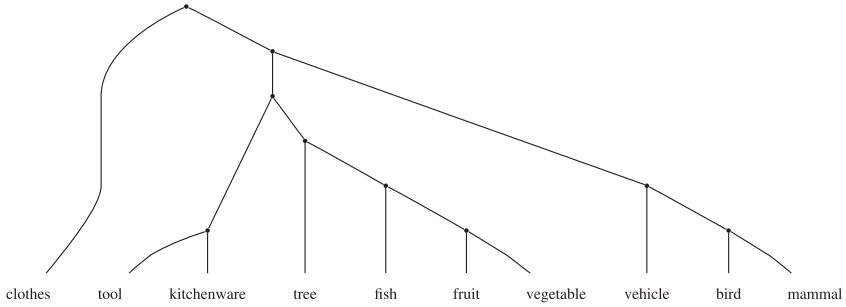
Finally, in Section 7 we noticed some problems with the nominally expressed function relation, due at least partially to the fact that the norms-based gold standard lumps together under this label what are really different properties. Here, it clearly emerges that the ''function'' cluster discovered by Strudel there (cluster 4 of Table 8) picks up at least one particular kind of functional property, that is, stuff whose containment or processing is the function of the concept—a definitional and distinctive characteristic of kitchenware, that is thus the only superordinate that displays this type of functional property.

8.2.2 Concept hierarchies

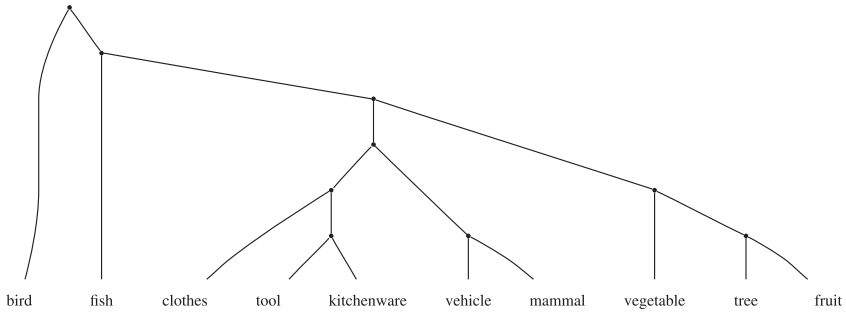
Fig. 1 shows the hierarchical clustering solution that CLUTO produced on top of the 10 flat clusters of the NORMS, Strudel, and DV models (the other CSMs produced very problematic hierarchical solutions, partly because some of their basic level clusters are not interpretable as they do not have a majority class). Each node in these trees dominates a set of



NORMS



Strudel



DV

Fig. 1. Hierarchical concept clustering of NORMS, Strudel, and DV models.

concept classes whose average internal pairwise similarity is higher than the one any of them has with any class not dominated by that node. Thus, for example, the third node from right in the NORMS tree tells us that the items in the fish cluster are on average closer to birds and mammals than to any other class.

The NORMS solution reproduces the classic three-way distinction into animals, plants, and artifacts (Caramazza & Shelton, 1998) as well as a two-way distinction between living and non living (Martin, 2007, and many others). However, there is an important mismatch with standard categorization in that vehicles are clustered with animals (into a ‘‘moving thing’’ superordinate, one supposes).

The Strudel solution features some closely knit higher-level superordinates we expected (tools grouped with kitchenware, vegetables and fruits, birds and mammals). It is also reproducing a three-way distinction into animals, plants, and tools, but with some interesting twists. Besides placing vehicles with animals (like NORMS), fish has shifted into the vegetable class, where they are actually closer to fruits and vegetables than trees are. Given that most fish in the test set are edible (*cod*, *salmon*, *trouts*, and *tuna*) and most of us only experience fish at the dinner table, what Strudel found is essentially an ''edible things'' class. Caramazza and Shelton originally proposed an evolutionary motivation for their general domains. The higher-level classes produced by Strudel—things that move on their own, things you eat, and things you use—make perfect sense in this perspective. Indeed, according to CLUTO the most descriptive property of the super-superordinate enclosing fish and plants is *eaten*, and for tools and kitchenware it is *used*. Given the largely ''utilitarian'' view of plants and fish displayed by Strudel, the algorithm does not group them with mammals and birds in a living class, but with the tools into a larger class of natural and artificial things you use in some way. Finally, clothes are an outlier category that Strudel fails to group with the others. All in all, Strudel's hierarchical categorization mixes patterns that follow received wisdom in cognitive science (basic identification of animals, plants, and artifacts) with intriguing generalizations that should be verified in humans (in particular, the three-way categorization into ''moving,'' ''edible,'' and ''usable'' as an alternative to the traditional animal, plant, and tool distinction). Once more, the differences between NORMS and Strudel correspond to different conceptualizations, rather than being a simple matter of wrong and right classification.

The DV model provides a good higher superordinate categorization of the (nonmoving) artifact and plant classes (within the latter, fruits pattern with trees rather than vegetables). However, it creates a spurious class of mammals and vehicles, that is then linked to the artifacts, and it fails to group fish and birds with other concept categories.

9. General discussion

Strudel is a new algorithm to extract structured concept descriptions from corpora of naturally occurring text. These descriptions are weighted lists of properties, with a surface indication (the ''type sketch'') of the relation they have with the concept. We showed that the property-based concept representations produced by Strudel are reasonable both qualitatively and in a quantitative comparison with speaker-generated descriptions. We showed, moreover, that Strudel property typing succeeds in the task of grouping properties into types according to the gold standard specification. Next, we showed that the descriptions produced by Strudel can be used as vectors to accomplish classic inter-concept similarity tasks of the sort performed by other CSMs, and that Strudel outperforms comparable CSMs in one of the most important similarity tasks, namely concept categorization into superordinates, a result which in turn supports the general program of adopting structured property representations as dimensions of CSMs. Properties and types also allowed Strudel to provide descriptions of the superordinates, and to classify some properties of the superordinate

concepts by type. Importantly, the same version of Strudel performed all the operations we described, with no need to go back to the training data for retuning. We think that this is a leap forward for CSMs, as being capable of maintaining different views of semantic relations is a basic feature of human conceptual cognition.

Strudel still requires serious improvements in many areas. The algorithm is not good at finding properties expressed by adjectives (because adjective–noun pairs tend not to occur in varied contexts), it does not collect properties expressed by more than one word (e.g., negated properties: *penguins do not fly*), and, most importantly, it does not handle polysemy. Moreover, Strudel should be compared with other CSMs, including topic models and BEAGLE, on a more varied battery of tasks, including tasks requiring a broader, associative or topical notion of similarity, where we might expect models that are less bound to a property-based level of concept representation to outperform Strudel. The performance of Strudel on other kinds of input, such as child-directed speech or encyclopedic texts—that might contain more generic properties of concepts—would also be interesting to assess.

We make the current version of the Strudel model, together with programs to extend it or retrain it on different input sources, fully available online with the Supporting Information. Strudel might be useful in language/knowledge engineering applications (e.g., as an aid to ontology population), as well as for further simulations of human conceptual tasks. In particular, Strudel can produce ''property norms'' that are not only easier to generate on a very large scale than the subject-elicited lists widely used for simulations and experimental design in psychology and cognitive science (Garrard et al., 2001; McRae et al., 2005; Vinson & Vigliocco, 2008), but they are in many ways complementary to the latter, as highlighted by our comparisons with the concept descriptions collected by McRae and colleagues.

Strudel tends to focus on activities and situations in which we interact with, use, and see things, whereas the McRae norms focus on descriptions of the physical properties of objects. This difference is reflected in categorization, where Strudel, going by the typical interactions we have with entities, classifies fish with fruit and vegetables (we cook and eat them all), and ends up with a two-way distinction between stuff we ''use'' (including what we eat) and stuff that moves on its own (see discussion in Section 8), instead of the traditional living–nonliving distinction emerging from the norms. We think that this is one of the most interesting findings of the experiments reported in this article. Paradoxically for a model based on computational analysis of text, Strudel turns out to have a rather ''embodied'' (Barsalou, 2008) view of the world. Given that lack of embodiment is sometimes seen as a serious flaw of CSMs (Glenberg & Robertson, 2000), an exciting direction for further research will be to explore to what extent a model like Strudel could deal with at least some of the criticism that is vented at CSMs from this perspective.

Notes

1. http://www.cs.ualberta.ca/~lindek/minipar.htm.
2. http://www.coli.uni-saarland.de/~pado/dv.html.

3. The norms are available from http://www.psychonomic.org/archive.
4. http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview.

Acknowledgments

We thank Raffaella Bernardi, Katrin Erk, Alessandro Lenci, and the *Cognitive Science* editor and reviewers for very useful feedback, as well as the developers of the tools and resources we used, especially Ken McRae and colleagues, James Van Overschelde and colleagues, George Karypis and colleagues, Sebastian Padó, Helmut Schmid, Stefan Evert, and Dekang Lin.

References

Almuhareb, A. (2006). *Attributes in lexical acquisition*. PhD dissertation. Essex, England: Department of Computing and Electronic Systems, University of Essex.

Almuhareb, A., & Poesio, M. (2004). Attribute-based and value-based clustering: An evaluation. *Proceedings of EMNLP*, 158–165.

Banko, M., Cafarella, M., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information extraction from the Web. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2670–2676.

Banko, M., & Etzioni, O. (2008). The tradeoffs between traditional and open relation extraction. *Proceedings of ACL*, 28–36.

Baroni, M., Bernardini, S., Ferraresi, A., & Zanchetta, E. (2009). The WaCky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, *43*(3), 209–231.

Baroni, M., & Lenci, A. (2008). Concepts and properties in word spaces. *Italian Journal of Linguistics*, *20*(1), 53–86.

Baroni, M., Lenci, A., & Onnis, L. (2007). ISA meets Lara: An incremental word space model for cognitively plausible simulations of semantic learning. *Proceedings of the ACL 2007 Workshop on Cognitive Aspects of Computational Language Acquisition*, 49–56.

Barsalou, L. (2008). Grounded cognition. *Annual Review of Psychology*, *59*, 617–645.

Blei, D., Ng, A., & Jordan, M. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Brachman, R. J., & Schmolze, J. G. (1985). An overview of the KL-ONE knowledge representation system. *Cognitive Science*, *9*(2), 171–216.

Capitani, E., Laiacona, M., Mahon, B., & Caramazza, A. (2003). What are the facts of semantic category-specific deficits? A critical review of the clinical evidence. *Cognitive Neuropsychology*, *20*, 213–261.

Caramazza, A., & Shelton, J. R. (1998). Domain-specific knowledge systems in the brain: The animate–inanimate distinction. *Journal of Cognitive Neuroscience*, *10*, 1–34.

Cimiano, P., & Wenderoth, J. (2005). Automatically learning qualia structures from the Web. *Proceedings of the ACL/SIGLEX Workshop on Deep Lexical Acquisition*, 28–37.

Curran, J., & Moens, M. (2002). Improvements in automatic thesaurus extraction. *Proceedings of the ACL Workshop on Unsupervised Lexical Acquisition*, 59–66.

Davidov, D., & Rappoport, A. (2008a). Classification of semantic relationships between nominals using pattern clusters. *Proceedings of ACL*, 227–235.

Davidov, D., & Rappoport, A. (2008b). Unsupervised discovery of generic relationships using pattern clusters and its evaluation by automatically generated SAT analogy questions. *Proceedings of ACL*, 692–700.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, *19*(1), 61–74.

Evert, S. (2005). *The statistics of word cooccurrences*. PhD dissertation. Stuttgart, Germany: IMS, Stuttgart University.

Fellbaum, C. (Ed). (1998). *WordNet: An electronic lexical database*. Cambridge, MA: The MIT Press.

Fodor, J. (1998). *Concepts: Where cognitive science went wrong*. Oxford, England: Oxford University Press.

Foltz, P., Kintsch, W., & Landauer, T. (1998). The measurement of textual coherence with latent semantic analysis. *Discourse Processes*, *25*, 285–307.

Garrard, P., Lambon Ralph, M., Hodges, J., & Patterson, K. (2001). Prototypicality, distinctiveness, and intercorrelation: Analyses of the semantic attributes of living and nonliving concepts. *Cognitive Neuropsychology*, *18*(2), 25–174.

Girju, R., Badulescu, A., & Moldovan, D. (2006). Automatic discovery of part-whole relations. *Computational Linguistics*, *32*(1), 83–135.

Glenberg, A., & Robertson, D. (2000). Symbol grounding and meaning: A comparison of high-dimensional and embodied theories of meaning. *Journal of Memory and Language*, *43*, 379–401.

Grefenstette, G. (1994). *Explorations in automatic thesaurus discovery*. Boston: Kluwer.

Griffiths, T., Steyvers, M., & Tenenbaum, J. (2007). Topics in semantic representation. *Psychological Review*, *114*, 211–244.

Hearst, M. (1992). Automatic acquisition of hyponyms from large text corpora. *Proceedings of COLING*, 539–545.

Hearst, M. (1998). Automated discovery of WordNet relations. In C. Fellbaum (Ed.), *WordNet: An electronic lexical database* (pp. 131–151). Cambridge, MA: The MIT Press.

Hofmann, Th. (2001). Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, *42*(1), 177–196.

Jackendoff, R. (1990). *Semantic structures*. Cambridge, MA: The MIT Press.

Jones, M., & Mewhort, D. (2007). Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, *114*, 1–37.

Karypis, G. (2003). CLUTO: A clustering toolkit. Technical report, 02-017, Department of Computer Science, University of Minnesota.

Laham, D. (1997). Latent semantic analysis approaches to categorization. *Proceedings of the 19th Cognitive Science Society Meeting*, 979.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, *104*(2), 211–240.

Levi, J. N. (1978). *The syntax and semantics of complex nominals*. New York: Academic Press.

Li, P., Burgess, C., & Lund, K. (2000). The acquisition of word meaning through global lexical co-occurrences. *Proceedings of the 31st Child Language Research Forum*, 167–178.

Lin, D. (1998). Automatic retrieval and clustering of similar words. *Proceedings of ACL*, 768–774.

Louwerse, M., Cai, Z., Hu, X., Ventura, M., & Jeuniaux, P. (2006). Cognitively inspired natural-language based knowledge representations: Further explorations of latent semantic analysis. *International Journal of Artificial Intelligence Tools*, *15*, 1021–1039.

Lund, K., & Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behaviour Research Methods*, *28*, 203–208.

Lund, K., Burgess, C., & Atchley, R. (1995). Semantic and associative priming in high dimensional semantic space. *Proceedings of the 17th Cognitive Science Society Meeting*, 660–665.

Manning, Ch., & Schütze, H. (1999). *Foundations of statistical natural language processing*. Cambridge, MA: The MIT Press.

Martin, A. (2007). The representation of object concepts in the brain. *Annual Review of Psychology*, *58*, 25–49.

Martin, A., & Chao, L. (2001). Semantic memory and the brain: Structure and process. *Current Opinions in Neurobiology*, *11*, 194–201.

McClelland, J., & Rogers, T. (2003). The parallel distributed processing approach to semantic cognition. *Nature Reviews Neuroscience*, *4*(4), 310–322.

McRae, K., Cree, G., Seidenberg, M., & McNorgan, C. (2005). Semantic feature production norms for a large set of living and nonliving things. *Behavior Research Methods*, *37*, 547–559.

McRae, K., de Sa, V. R., & Seidenberg, M. S. (1997). On the nature and scope of featural representations of word meaning. *Journal of Experimental Psychology*, *126*(2), 99–130.

Moscoso del Prado, M., & Sahlgren, M. (2002). An integration of vector-based semantic analysis and simple recurrent networks for the automatic acquisition of lexical representations from unlabeled corpora. *Proceedings of the Linguistic Knowledge Acquisition and Representation Workshop*, 71–80.

Murphy, G. (2002). *The big book of concepts*. Cambridge, MA: The MIT Press.

Norman, D. A., Rumelhart, D. E., & the LNR Research Group. (1975). *Explorations in cognition*. San Francisco: W. H. Freeman.

Padó, S., & Lapata, M. (2007). Dependency-based construction of semantic space models. *Computational Linguistics*, *33*(2), 161–199.

Pantel, P., & Pennacchiotti, M. (2006). Espresso: Leveraging generic patterns for automatically harvesting semantic relations. *Proceedings of COLING/ACL*, 113–120.

Plaut, D. (1995). Semantic and associative priming in a distributed attractor network. *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society*, 37–42.

Plaut, D. C., & Shallice, T. (1993). Deep dyslexia: A case study of connectionist neuropsychology. *Cognitive Neuropsychology*, *10*, 377–500.

Poesio, M., & Almuhareb, A. (2005). Identifying concept attributes using a classifier. *Proceedings of the ACL Workshop on Deep Lexical Semantics*, 18–27.

Poesio, M., & Almuhareb, A. (2008). Extracting concept descriptions from the Web: The importance of attributes and values. In P. Buitelaar & P. Cimiano (Eds.), *Bridging the gap between text and knowledge* (pp. 29–44). Amsterdam, The Netherlands: IOS Press.

Poesio, M., Baroni, M., Murphy, B., Barbu, E., Lombardi, L., Almuhareb, A., Vigliocco, G., & Vinson, D. (2007) Speaker-generated and corpus-generated concept features. Oral presentation at *Concept types and frames in language, cognition, and science*, Düsseldorf, Germany, August 20, 2007.

Pustejovsky, J. (1995). *The generative lexicon*. Cambridge, MA: The MIT Press.

Rapp, R. (2003). Word sense discovery based on sense descriptor dissimilarity. *Proceedings of the Ninth Machine Translation Summit*, 315–322.

Rapp, R. (2004). A freely available automatically generated thesaurus of related words. *Proceedings of LREC*, 395–398.

Rogers, T., & McClelland, J. (2004). *Semantic cognition: A parallel distributed processing approach*. Cambridge, MA: The MIT Press.

Rosch, E. (1975). Cognitive representation of semantic categories. *Journal of Experimental Psychology: General*, *104*, 192–233.

Rowland, C., Pine, J., Lieven, E., & Theakston, A. (2005). The incidence of error in young children's wh-questions. *Journal of Speech, Language and Hearing Research*, *48*(2), 384–404.

Rumelhart, D., McClelland, J., & the PDP Research Group. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition*. Cambridge, MA: The MIT Press.

Sahlgren, M. (2006). *The word-space model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. PhD dissertation. Stockholm, Sweden: Department of Linguistics, Stockholm University.

Schulte im Walde, S. (2008). *Theoretical adequacy, human data and classification approaches in modelling word properties, word relatedness and word classes* (Habilitation). Saarbücken, German: School of Philosophy, Saarland University.

Schunn, C. D. (1999). The presence and absence of category knowledge in LSA. *Proceedings of the 21st Annual Conference of the Cognitive Science Society*, 643–648.

Schütze, H. (1997). *Ambiguity resolution in language learning*. Stanford, CA: CSLI.

Steinberger, J., Poesio, M., Kabadjov, M., & Jezek, K. (2007). Two uses of anaphora resolution in summarization. *Information Processing and Management*, *43*, 1663–1680.

Van Overschelde, J., Rawson, K., & Dunlosky, J. (2004). Category norms: An updated and expanded version of the Battig and Montague (1969) norms. *Journal of Memory and Language*, *50*, 289–335.

Vigliocco, G., Vinson, D., Lewis, W., & Garrett, M. (2004). Representing the meanings of object and action words: The featural and unitary semantic space hypothesis. *Cognitive Psychology*, *48*, 422–488.

Vinson, D., & Vigliocco, G. (2008). Semantic feature production norms for a large set of objects and events. *Behavior Research Methods*, *40*(1), 183–190.

Woods, W. A. (1975). What's in a link: Foundations for semantic networks. In D. G. Bobrow & A. M. Collins (Eds.), *Representation and understanding: Studies in cognitive science* (pp. 35–82). New York: Academic Press.

Zhao, Y., & Karypis, G. (2001). Criterion functions for document clustering: Experiments and analysis. Technical report, 01-40, Department of Computer Science, University of Minnesota.

Supporting Information

Additional Supporting Information may be found in the online version of this article on Wiley InterScience:

Supplemental Materials.

Please note: Wiley-Blackwell is not responsible for the content or functionality of any supporting materials supplied by the authors. Any queries (other than missing material) should be directed to the corresponding author for the article.