Quantificational features in distributional word representations

Tal Linzen¹,²  Emmanuel Dupoux¹  Benjamin Spector²
¹Laboratoire de Sciences Cognitives et Psycholinguistique  ²Institut Jean Nicod
École Normale Supérieure
PSL Research University
{tal.linzen, benjamin.spector}@ens.fr
emmanuel.dupoux@gmail.com

Abstract

Do distributional word representations encode the linguistic regularities that theories of meaning argue they should encode? We address this question in the case of the logical properties (monotonicity, force) of quantificational words such as everything (in the object domain) and always (in the time domain). Using the vector offset approach to solving word analogies, we find that the skip-gram model of distributional semantics behaves in a way that is remarkably consistent with encoding these features in some domains, with accuracy approaching 100%, especially with medium-sized context windows. Accuracy in other domains was less impressive. We compare the performance of the model to the behavior of human participants, and find that humans performed well even where the models struggled.

1 Introduction

Vector-space models of lexical semantics (VSMs) represent words as points in a high-dimensional space. Similar words are represented by points that are close together in the space. VSMs are typically trained on a corpus in an unsupervised way; the goal is for words that occur in similar contexts to be assigned similar representations. The context of a word in a corpus is often defined as the set of words that occur in a small window around the word of interest (Lund and Burgess, 1996; Turney and Pantel, 2010). VSM representations have been shown to be useful in improving the performance of NLP systems (Turian et al., 2010; Bansal et al., 2014) as well as in predicting cognitive measures such as similarity judgments and semantic priming (Jones et al., 2006; Hill et al., 2015).

While there is evidence that VSM representations encode useful information about the meaning of open-class words such as dog or table, less is known about the extent to which they capture abstract linguistic properties, in particular the aspects of word meaning that are crucial in logical reasoning. Some have conjectured that those properties are unlikely to be encoded in VSMs (Lewis and Steedman, 2013), but evidence that VSMs encode features such as syntactic category or verb tense suggests that this pessimism is premature (Mikolov et al., 2013c; Levy and Goldberg, 2014).

The goal of this paper is to evaluate to what extent logical features are encoded in VSMs. We undertake a detailed analysis of words with quantificational features, such as everybody or nowhere. To assess whether a particular linguistic feature is encoded in a vector space, we adopt the vector offset approach to the analogy task (Turney, 2006; Mikolov et al., 2013c; Dunbar et al., 2015). In the analogy task, a system is requested to fill in the blank in a sentence:

(1) man is to woman as king is to ___.

The system is expected to infer the relation between the first two words—man and woman—and find a word that stands in the same relation to king. When this task is solved using the offset method, there is no explicit set of relations that the system is trained to identify. We simply subtract the vector for man from the vector for woman and add it to king. If the offset woman — man represents an abstract gender feature, adding that offset to king should lead us to queen (Figure 1).

In the rest of this paper, we describe the set of analogy problems that we used to evaluate the VSMs’ representation of quantificational features, and explore how accuracy is affected by the con-
text windows used to construct the VSM. We then report two experiments that examine the robustness of the results. First, we determine whether the level of performance that we expect from the VSMs is reasonable, by testing how well humans solve the same analogy problems. Second, we investigate how the quality of the representations is affected by the size of the training corpus.

A large and constantly expanding range of VSM architectures have been proposed in the literature (Mikolov et al., 2013a; Pennington et al., 2014; Turney and Pantel, 2010). Instead of exploring the full range of architectures, the present study will focus on the skip-gram model, implemented in word2vec (Mikolov et al., 2013b). This model has been argued to perform either better than or on a par with competing architectures, depending on the task and on hyperparameter settings (Baroni et al., 2014; Levy et al., 2015). Particularly pertinent to our purposes, Levy et al. (2015) find that the skip-gram model tends to recover formal linguistic features more accurately than traditional distributional models.

2 Quantificational words

We focus on words that quantify over the elements of a domain, such as everybody or nowhere. We restrict our attention to single words that include the domain of quantification as part of their meaning – that is, we exclude determiners (every) and phrases (every person). The meaning of a quantifier is determined by three factors: quantificational force, polarity and domain of quantification. We describe these factors in turn.

2.1 Quantificational force

We focus on universal and existential quantificational words, which can be translated into first-order logic using a universal (∀) or existential (∃) quantifier. For example, everybody and nobody are both universal:

\[
\text{everybody} \quad \text{something} \quad \text{nothing} \\
\text{everywhere} \quad \text{somewhere} \quad \text{nowhere} \\
\text{always} \quad \text{sometimes} \quad \text{never} \\
\text{must} \quad \text{can} \quad \text{cannot} \\
\text{require} \quad \text{allow} \quad \text{forbid}
\]

<table>
<thead>
<tr>
<th>INC.</th>
<th>Existential</th>
<th>DEC.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Universal</td>
<td>Universal</td>
<td></td>
</tr>
<tr>
<td>PERSON</td>
<td>everybody</td>
<td>somebody</td>
</tr>
<tr>
<td>OBJECT</td>
<td>everything</td>
<td>something</td>
</tr>
<tr>
<td>PLACE</td>
<td>everywhere</td>
<td>somewhere</td>
</tr>
<tr>
<td>TIME</td>
<td>always</td>
<td>sometimes</td>
</tr>
<tr>
<td>MODAL</td>
<td>must</td>
<td>can</td>
</tr>
<tr>
<td>MODAL V.</td>
<td>require</td>
<td>allow</td>
</tr>
</tbody>
</table>

Table 1: All of the words tested in the experiments (INC = Increasing, DEC = Decreasing).

(2) Everybody smiles:  
∀x.person(x) → smiles(x)

(3) Nobody smiles:  
∀x.person(x) → ¬smiles(x)

Somebody is existential:

(4) Somebody smiles:  
∃x.person(x) ∧ smiles(x)

English has quantificational expressions that don’t fall into either category (three people, most things). Those are usually not encoded as a single English word, and are therefore not considered in this paper.

2.2 Polarity

Quantifiers that can be expressed as a single word are in general either increasing or decreasing. A quantifier is increasing if any predicate that is true of the quantifier can be broadened without affecting the truth value of the sentence (Barwise and Cooper, 1981). For example, since everybody is increasing, (5-a) entails (5-b):

(5) a. Everybody went out to a death metal concert last night.
   b. Everybody went out last night.

By contrast, in decreasing quantifiers such as nobody the truth of broader predicates entails the truth of narrower ones:

(6) a. Nobody went out last night.
   b. Nobody went out to a death metal concert last night.

2.3 Domain

We studied six domains. The first three domains are intuitively straightforward: PERSON (e.g., everybody); OBJECT (e.g., everything); and PLACE
(e.g., everywhere). The three additional domains are described below.

**TIME:** Temporal adverbs such as always and seldom are naturally analyzed as quantifying over situations or events (Lewis, 1975; de Swart, 1993). The sentence Caesar always awoke before dawn, for example, can be seen as quantifying over waking events and stating that each of those events occurred before dawn.

**MODAL:** Modal auxiliaries such as must or can quantify over relevant possible worlds (Kripke, 1959). Consider, for example, the following sentences:

\[(7) \quad \begin{align*}
    a. & \quad \text{Anne must go to bed early.} \\
    b. & \quad \text{Anne can go to bed early.}
\end{align*}
\]

Assuming deontic modality, such as the statement of a rule, (7-a) means that in all worlds in which the rule is obeyed, Anne goes to bed early, whereas (7-b) means that there exists at least one world consistent with the speaker’s orders in which she goes to bed early.

**MODAL VERB:** Verbs such as request and forbid can be paraphrased using modal auxiliaries: he allowed me to stay up late is similar in meaning to he said I can stay up late. It is plausible to argue that allow is existential and increasing, just like can.

### 3 Evaluation

In what follows, we use the following notation (Levy and Goldberg, 2014):

\[(8) \quad a : a^* :: b : __
\]

The offset model is typically understood as in Figure 1: the analogy task is solved by finding \( x = a^* - a + b \). In practice, since the space is continuous, \( x \) is unlikely to precisely identify a word in the vocabulary. The guess is then taken to be the word \( x^* \) that is nearest to \( x \):

\[
x^* = \arg \max_{x'} \cos(x', a^* - a + b) \tag{1}
\]

where \( \cos \) denotes the cosine similarity between the vectors. This point has a significant effect on the results of the offset method, as we will see below. Following Mikolov et al. (2013c) and Levy and Goldberg (2014), we normalize \( a \), \( a^* \) and \( b \) prior to entering them into Equation 1.

**Trivial responses:** \( x^* \) as defined above is almost always trivial: in our experiments the nearest neighbor of \( x \) was either \( a^* \) (11% of the time) or \( b \) (88.9% of the time). Only in a single analogy out of the 2160 we tested was it not one of those two options. Following Mikolov et al. (2013c), then, our guess \( x^* \) will be the nearest neighbor of \( x \) that is not \( a, a^* \) or \( b \).

**Baseline:** The fact that the nearest neighbor of \( a^* - a + b \) tends to be \( b \) itself suggests that \( a^* - a \) is typically small in comparison to the distance between \( b \) and any of its neighbors. Even if \( b \) is excluded as a guess, then, one might be concerned that the analogy target \( b^* \) is closer to \( b \) than any of its neighbors. If that is the case, our success on the analogy task would not be informative: our results would stay largely the same if \( a^* - a \) were replaced by a random vector of the same magnitude (Linzen, 2016). To address this concern, we add a baseline that solves the analogy task by simply returning the nearest neighbor of \( b \), ignoring \( a \) and \( a^* \) altogether.

**Multiplication:** Levy and Goldberg (2014) point out that the word \( x^* \) that is closest to \( a^* - a + b \) in terms of cosine similarity is the one that maximizes the following expression:

\[
    \arg \max_{x'} \cos(x', a^*) - \cos(x', a) + \cos(x', b)) \tag{2}
\]

They report that replacing addition with multiplication improves accuracy on the analogy task:

\[
    \arg \max_{x'} \frac{\cos(x', a^*) \cos(x', b)}{\cos(x', a)} \tag{3}
\]

We experiment with both methods.

**Synonyms:** Previous studies required an exact match between the guess and the analogy target selected by the experimenter. This requirement may underestimate the extent to which the space encodes linguistic features, since the bundle of semantic features expressed by the intended target can often be expressed by one or more other words. This is the case for everyone and everybody, prohibit and forbid or can’t and cannot. As such, we considered synonyms of \( b^* \) to be exact matches. Likewise, we considered synonyms of \( a \), \( a^* \) and \( b \) to be trivial responses and excluded them from consideration as guesses.

This treatment of synonyms is reasonable when the goal is to probe the VSM’s semantic representations (as it often is), but may be inappropriate for
other purposes. If, for example, the analogy task is used as a method for generating inflected forms, prohibiting would not be an appropriate guess for like: liking :: forbid: __.

Partial success metrics: We did not restrict the guesses to words with quantificational features: all of the words in the vocabulary, including words like penguin and melancholy, were potential guesses. In addition to counting exact matches \((x^* = b^*)\), then, we keep track of the proportion of cases in which \(x^*\) was a quantificational word in one of the six relevant domains.

Within the cases in which \(x^*\) was a quantificational word, we separately counted how often \(x^*\) had the expected domain, the expected polarity and the expected force. To be able to detect such partial matches, we manually added some words to our vocabulary that were not included in the set in Table 1. These included items starting with any, such as anywhere or anybody, as well as additional temporal adverbs (seldom, often).

Finally, we record the rank of \(b^*\) among the 100 nearest neighbors of \(x\), where a rank of 1 indicates an exact match. It was often the case that \(b^*\) was not among the 100 nearest neighbors of \(x\); we therefore record how often \(b^*\) was ranked at all.

4 Experimental setup

4.1 Analogies

For each ordered pair of domains \((6 \times 5 = 30\) pairs in total), we constructed all possible analogies where \(a\) and \(a^*\) were drawn from one domain (the source domain) and \(b\) and \(b^*\) from the other (the target domain). Since there are three words per domain, we had six possible analogies per domain pair, for a total of 180 analogies.

Each set of four words was used to construct multiple analogies. Those analogies are in general not equivalent. For example, the words everybody, nobody, everywhere and nowhere make up the following analogies:

\[
\begin{align*}
(9) \quad & \text{everybody : nobody :: everywhere : __} \\
(10) \quad & \text{nobody : everybody :: nowhere : __} \\
(11) \quad & \text{everywhere : nowhere :: everybody : __} \\
(12) \quad & \text{nowhere : everywhere :: nobody : __}
\end{align*}
\]

The neighborhoods of everywhere and nobody may well differ in density. Since the density of the neighborhood of \(b\) affects the results of the offset method, the result is not invariant to a permutation of the words in an analogy. It is, however, invariant to replacing a within-domain analogy with an across-domain one. The following analogy is equivalent to (9):

\[(13) \quad \text{everybody : everywhere :: nobody : __} \]

This analogy would be solved by finding the nearest neighbor of everywhere−everybody+nobody, which is, of course, the same as the nearest neighbor of nobody−everybody+everywhere used to solve (9). We do not include such analogies.

4.2 VSMs

We trained our VSMs using the skip-gram with negative sampling algorithm implemented in hyper words,\(^1\) which extends word2vec to allow finer control over hyperparameters. The vectors were trained on a concatenation of ukWaC (Baroni et al., 2009) and a 2013 dump of the English Wikipedia, 3.4 billion words in total.

The skip-gram model has a large number of parameters. We set most of those parameters to values that have been previously shown to be effective (Levy et al., 2015); we list those values below. We only vary three parameters that control the context window. Syntactic category information has been shown to be best captured by narrow context windows that encode the position of the context word relative to the focus word (Redington et al., 1998; Sahlgren, 2006). Our goal in varying these parameters is to identify the contexts that are most conducive to recovering logical information.

Window size: We experimented with context windows of 2, 5 or 10 words on either side of the focus word (i.e., a window of size 2 around the focus word consists of four context words).

Window type: When constructing the vector space, the skip-gram model performs frequency-based pruning: rare words are discarded in all cases and very frequent words are discarded probabilistically. We experimented with static and dynamic windows. The size of static windows is determined prior to frequency-based word deletion. By contrast, the size of dynamic windows is determined after frequent and infrequent words are deleted. This means that dynamic windows often include words that are farther away from the focus words than the nominal window size, and

\(^1\)https://bitbucket.org/omerlevy/hyperwords
that words that tend to have very frequent function words around them will systematically have a larger effective context window.

**Context type:** We experimented with bag-of-words (nonpositional) contexts and positional contexts. In nonpositional contexts, a context word cat is treated in the same way regardless of its distance from the focus word and of whether it follows or precedes it. In positional contexts, on the other hand, context words are annotated with their position relative to the focus words; the context word cat\(^{-2}\) is considered to be distinct from cat\(^{+1}\).

**Fixed hyperparameters:** We used the following values for the rest of the hyperparameters: 500-dimensional words vectors; 15 negative samples per focus word; words with a frequency of less than 100 were discarded; words with unigram probability above 10\(^{-5}\) were probabilistically discarded (preliminary experiments showed that a 10\(^{-3}\) threshold reduced performance across the board); negative samples were drawn from the unigram frequency distribution, after that distribution was smoothed with exponent \(\alpha = 0.75\); we performed one iteration through the data.

### Table 2: Results on all hyperparameter settings, evaluated using three methods: B(aseline), Offset and M(ultiplication).

<table>
<thead>
<tr>
<th>Size</th>
<th>Context</th>
<th>Window</th>
<th>B</th>
<th>O</th>
<th>M</th>
<th>O - B</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Nonpos</td>
<td>Dynamic</td>
<td>.08</td>
<td>.32</td>
<td>.34</td>
<td>.24</td>
</tr>
<tr>
<td>2</td>
<td>Nonpos</td>
<td>Static</td>
<td>.06</td>
<td>.23</td>
<td>.24</td>
<td>.17</td>
</tr>
<tr>
<td>2</td>
<td>Pos</td>
<td>Dynamic</td>
<td>.06</td>
<td>.29</td>
<td>.32</td>
<td>.24</td>
</tr>
<tr>
<td>2</td>
<td>Pos</td>
<td>Static</td>
<td>.06</td>
<td>.24</td>
<td>.27</td>
<td>.19</td>
</tr>
<tr>
<td>5</td>
<td>Nonpos</td>
<td>Dynamic</td>
<td>.07</td>
<td>.28</td>
<td>.29</td>
<td>.22</td>
</tr>
<tr>
<td>5</td>
<td>Nonpos</td>
<td>Static</td>
<td>.11</td>
<td>.35</td>
<td>.36</td>
<td>.24</td>
</tr>
<tr>
<td>5</td>
<td>Pos</td>
<td>Dynamic</td>
<td>.03</td>
<td>.29</td>
<td>.31</td>
<td>.27</td>
</tr>
<tr>
<td>5</td>
<td>Pos</td>
<td>Static</td>
<td>.06</td>
<td>.28</td>
<td>.29</td>
<td>.23</td>
</tr>
<tr>
<td>10</td>
<td>Nonpos</td>
<td>Dynamic</td>
<td>.08</td>
<td>.28</td>
<td>.29</td>
<td>.19</td>
</tr>
<tr>
<td>10</td>
<td>Nonpos</td>
<td>Static</td>
<td>.17</td>
<td>.31</td>
<td>.31</td>
<td>.14</td>
</tr>
<tr>
<td>10</td>
<td>Pos</td>
<td>Dynamic</td>
<td>.17</td>
<td>.32</td>
<td>.31</td>
<td>.16</td>
</tr>
<tr>
<td>10</td>
<td>Pos</td>
<td>Static</td>
<td>.11</td>
<td>.26</td>
<td>.26</td>
<td>.15</td>
</tr>
</tbody>
</table>

**Offset method:** Overall accuracy was fairly low (mean: 0.29, range: 0.23 – 0.35), somewhat lower than the 0.4 accuracy that Mikolov et al. (2013c) report for their syntactic features.\(^2\) Strikingly, \(b^*\) was among the 100 nearest neighbors of \(x\) only in 70% of the cases. When the guess was a quantificational word (61% of the time), it was generally in the right domain (93%). Its polarity was correct 72% of the time, and its force 54% of the time.

The static nonpositional 5-word VSM achieved the best accuracy (35%), best average rank (5.5) and was able to recover the most quantificational features (polarity: 82% correct; force: 63% correct; both proportions are conditioned on the guess being a quantificational word).

**Alternatives to the offset method:** In line with the results reported by Levy and Goldberg (2014), we found that substituting multiplication for addition resulted in slightly improved performance in 10 out of 12 VSMs, though the improvement in each individual VSM was never significant according to Fisher’s exact test (Table 2). If we take each VSM to be an independent observation, the difference across all VSMs is statistically significant in a t-test \((t = 2.45, p = 0.03)\).

The baseline that ignores \(a\) and \(a^*\) altogether reached an accuracy of up to 0.17, sometimes accounting for more than half the accuracy of the offset method. The success of the baseline is significant, given that chance level is very low (recall that all but the rarest words in the corpus were possible guesses). Still, the offset method was significantly more accurate than the baseline in all VSMs \((10^{-12} < p < 0.003, \text{Fisher’s exact test})\).

**Differences across domains:** We examine the performance of the offset method in the best-performing VSM in greater detail. There were dramatic differences in accuracy across target domains. When \(b^*\) was a PERSON, guesses were correct 73% of the time; the correct guess was one of the top 100 neighbors 87% of the time, and its average rank was 1.31. Conversely, when \(b^*\) was a MODAL VERB, the guess was never correct; in fact, in this target domain, \(b^*\) was one of the 100 nearest neighbors of \(x\) only 7% of the time, and the average rank in these cases was 59 (see Table

\(^2\)Note that the figure reported by Mikolov et al. (2013c) collapses across several different types of syntactic features, some of which are encoded with accuracy higher than 0.4 and some with lower accuracy (Levy and Goldberg, 2014; Linzen, 2016).
Table 3: A sample of errors made by the [5, Nonpositional, Static] VSM (an error is an analogy problem where the correct answer was not the nearest neighbor of \( x = a^* - a + b \)). Four analogies are shown per target domain; \( x^*_1 \), \( x^*_2 \) and \( x^*_3 \) are the nearest, second nearest and third nearest neighbors of \( x \), respectively. The rank is marked as n/a the correct answer was not one of the 100 nearest neighbors of \( x \).

Effect of context parameters: Overall, the influence of context parameters on accuracy was not dramatic. When the VSMs are compared based on the extent that the offset method improves over the baseline (\( O - B \) in Table 2), a somewhat clearer picture emerges: the improvement is greatest in intermediate window sizes, either 5-word windows or dynamic 2-word windows. This contrasts with findings on the acquisition of syntactic categories, where narrower contexts performed best (Redington et al., 1998), suggesting that the cues to quantificational features are further from the focus word than cues to syntactic category.

One candidate for such a cue is the word’s compatibility with negative polarity items (NPI) such as any. NPIs are often licensed by decreasing quantifiers (Fauconnier, 1975): nobody ate any cheese is grammatical, but *everybody ate any cheese isn’t. Whereas contextual cues to syntactic category—e.g., the before nouns—are often directly adjacent to the focus word, any will typically be part of a different constituent from the focus word, and is therefore quite likely to fall outside a narrow context window.

We did not find a systematic effect of the type of context (position vs. nonposition). However, as Section 7 below shows, this parameter does affect performance when the VSMs are trained on smaller corpora.
Figure 2: On the left: accuracy of the best model (static nonpositional 5-word context), broken down by source (in the y-axis) and target (in the x-axis) domain. On the right: human responses.

6 How well do humans do the task?

Some of the analogies are intuitively fairly difficult: quantification over possible deontic worlds (require vs. forbid) is quite different from quantification over individuals (everybody vs. nobody). Those are precisely the domains in which the VSMs performed poorly. Are we asking too much of our VSM representations? Can humans perform this task?

To answer this question, we gave the same analogies to human participants recruited through Amazon Mechanical Turk. We divided our 180 quantificational analogies into five lists of 36 analogies each. Each list additionally contained four practice trials presented in the beginning of the list and ten catch trials interspersed throughout the list. These additional trials contained simple analogies, such as big : bigger :: strong : __ or brother : sister :: son : __. Each of the lists was presented to ten participants (50 participants in total). They were asked to type in a word that had the same relationship to the third word as the first two words had to each other.

We excluded participants that made more than three mistakes on the catch trials (three participants) as well as one participant who did not provide any answer to some of the questions. While mean accuracy varied greatly among subjects (range: 0.22 – 1; mean: 0.68; median: 0.69; standard deviation: 0.17), it was in general much higher than the accuracy of the VSMs.

Figure 2b presents the human participants’ aver-
age accuracy by source and target domain. Mean accuracy was 0.45 or higher for all combinations of source and target domains. Logistic regression confirmed that having MODAL VERB and MODAL as either the source or target domain led to lower accuracy. There were no statistically significant differences between those two domains or among the remaining four domains, with the exception of TIME as a target domain, which was less accurate than PLACE, OBJECT and PERSON.

The VSMs did not have access to the morphological structure of the words. This makes the comparison with humans difficult: it is hard to see how human participants could be stopped from accessing that knowledge when performing an analogy such as nowhere : somewhere :: nobody : __. Notably, however, the difference in performance between the morphologically marked domains and the other domains is if anything more marked in the VSMs than in humans. Moreover, there is a fairly small difference in the accuracy of our human participants between PLACE and TIME as target domains, even though the former is morphologically marked and the latter isn’t.

7 Effect of training corpus size

The VSMs trained on our 3.4 billion token corpus achieved very good performance on the analogy task, at least in some of the domains. How dependent is the performance of the models on the size of the training corpus? To address this question, we sampled four subcorpora from our Wikipedia corpus, with 100K, 1M, 3M and 10M sentences. As the average sentence length in the corpus is 18 words, the corpora contained 1.8M, 18M, 54M and 180M tokens, respectively.

Given that VSM accuracy was low in some of the domains even when the spaces were trained on 3.4G tokens, we limit our experiments in this section to the OBJECT and PERSON domains. We made two changes to the hyperparameters settings that were not modulated in the VSMs trained on the full corpus. First, we lowered the threshold for rare word deletion (100K / 1M sentences: 10; 3M sentences: 50; 10M sentences: 100). Second, we experimented with smaller vectors (100, 300 and 500), under the assumption that it may be more difficult to train large vectors on a small data set. As before, we experimented with window sizes of 2, 5 and 10 words on either side of the focus word and with positional and nonpositional contexts. The size of the windows was always static.

Figure 3 shows the accuracy of the analogy task averaged across vector sizes and window sizes. VSMs trained on the 100K and 1M subcorpora completely failed to perform the task: with the exception of one model that performed one out of the 12 analogies correctly, accuracy was always 0. The VSMs trained on the 3M and 10M sentences subcorpora perform better (between 0.27 and 0.39 on average), though still much worse than the VSMs trained on the full corpus. The type of context had a large effect on the success of the model: VSMs with positional contexts trained on the 3M subcorpus had extremely low accuracy, whereas on the 10M subcorpus positional contexts performed better than nonpositional ones. The performance advantage of positional contexts was larger on the 10M corpus than on the full corpus.

Hart and Risley (1995) estimate that American children are exposed to between 3 and 11 million words every year, depending on the socioeconomic status of their family. The 1M and 3M sentence corpora therefore represent plausible amounts of exposure for a child; the adults tested in Section 6 may have seen the equivalent of 10M sentences. The degraded performance of the VSMs on these smaller training corpora suggests that distributional information alone is unlikely to be sufficient for humans’ acquisition of quantification, and that an adequate cognitive model would need to consider richer types of context, such as syntactic context and discourse structure, or to make explicit reference to the way these words are used in logical reasoning.
8 Related work

There is a large body of work on the evaluation of VSMs (Turney and Pantel, 2010; Hill et al., 2015). A handful of recent papers have looked at distributional representations of logical words. Baroni et al. (2012) extracted corpus-based distributional representations for quantifier phrases such as all cats and no dogs, and trained a classifier to detect entailment relations between those phrases; for example, the classifier might learn that all cats entails some cats. Bernardi et al. (2013) introduce a phrase similarity challenge that relies on the correct interpretation of determiners (e.g., orchestra is expected to be similar to many musicians), and use it to evaluate VSMs and composition methods. Hermann et al. (2013) discuss the difficulty of accounting for negation in a distributional semantics framework.

Another line of work seeks to combine the graded representations of content words such as manual or book with a symbolic representation of logical words (Garrette et al., 2014; Lewis and Steedman, 2013; Herbelot and Vecchi, 2015). Our work, which focuses on the quality of graded representation of logical words, can be seen as largely orthogonal to this line of work.

Finally, our study is related to recent neural network architectures designed to recognize entailment and other logical relationships between sentences (Bowman et al., 2015; Rocktäschel et al., 2016). Those systems learn word vector representations that are optimized to perform an explicit entailment task (when trained in conjunction with a compositional component). In future work, it may be fruitful to investigate whether those representations encode logical features more faithfully than the unsupervised representations we experimented with.

9 Conclusion

The skip-gram model, like earlier models of distributional semantics, represents words in a vector space using only their bag-of-words contexts in a corpus. We tested whether the representations that this model acquires for words with quantificational content encode the logical features that theories of meaning predict they should encode. We addressed this question using the offset method for solving the analogy task, $a : a^* :: b : ___$ (e.g., everyone : someone :: everywhere : ___).

Distributional methods successfully recovered quantificational features in many cases. Accuracy was higher when the context window was of an intermediate size, sometimes approaching 100% on simpler domains. Performance on other domains was poorer, however. Humans given the same task also showed variability across domains, but achieved better accuracy overall, suggesting that there is room for improving the VSMs. Finally, we showed that the VSMs require large amounts of training data to perform the task well, suggesting that the simplest form of distributional learning is not sufficient for acquiring logical features given the amount of language input that humans are exposed to.

Acknowledgements

We thank Marco Baroni, Emmanuel Chemla, Anne Christophe and Omer Levy for comments and technical assistance. This research was supported by the European Research Council (grant ERC-2011-AdG 295810 BOOTPHON) and the Agence Nationale pour la Recherche (grants ANR-10-IDEX-0001-02 PSL and ANR-10-LABX-0087 IEC).

References


