Issues in evaluating semantic spaces using word analogies

Tal Linzen
LSCP & IJN
École Normale Supérieure
PSL Research University
tal.linzen@ens.fr

Abstract

The offset method for solving word analogies has become a standard evaluation tool for vector-space semantic models: it is considered desirable for a space to represent semantic relations as consistent vector offsets. We show that the method's reliance on cosine similarity conflates offset consistency with largely irrelevant neighborhood structure, and propose simple baselines that should be used to improve the utility of the method in vector space evaluation.

1 Introduction

Vector space models of semantics (VSMs) represent words as points in a high-dimensional space (Turney and Pantel, 2010). There is considerable interest in evaluating VSMs without needing to embed them in a complete NLP system. One such intrinsic evaluation strategy that has gained in popularity in recent years uses the offset approach to solving word analogy problems (Levy and Goldberg, 2014; Mikolov et al., 2013c; Mikolov et al., 2013a; Turney, 2012). This method assesses whether a linguistic relation — for example, between the base and gerund form of a verb (*debug* and *debugging*) — is consistently encoded as a particular linear offset in the space. If that is the case, estimating the offset using one pair of words related in a particular way should enable us to go back and forth between other pairs of words that are related in the same way, e.g., *scream* and *screaming* in the base-to-gerund case (Figure 1).

Since VSMs are typically continuous spaces, adding the offset between *debug* and *debugging* to *scream* is unlikely to land us exactly on any particular word. The solution to the analogy problem is therefore taken to be the word closest in cosine similarity to the landing point. Formally, if the analogy is given by

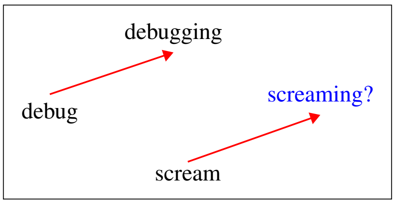

Figure 1: Using the vector offset method to solve the analogy task (Mikolov et al., 2013c).

$$a : a^* :: b : \_\_\_ \qquad (1)$$

where in our example a is *debug*, a^* is *debugging* and b is *scream*, then the proposed answer to the analogy problem is

$$x^* = \operatorname*{argmax}_{x'} \cos(x', a^* - a + b) \qquad (2)$$

where

$$\cos(v, w) = \frac{v \cdot w}{\|v\|\|w\|} \qquad (3)$$

The central role of cosine similarity in this method raises the concern that the method does not only evaluate the consistency of the offsets $a^* - a$ and $b^* - b$ but also the neighborhood structure of $x = a^* - a + b$. For instance, if a^* and a are very similar to each other (as *scream* and *screaming* are likely to be) the nearest word to x may simply be the nearest neighbor of b. If in a given set of analogies the nearest neighbor of b tends to be b^*, then, the method may give the correct answer regardless of the consistency of the offsets (Figure 2).

In this note we assess to what extent the performance of the offset method provides evidence for offset consistency despite its potentially problematic reliance on cosine similarity. We use two

Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NLP, pages 13–18,
Berlin, Germany, August 12, 2016. ©2016 Association for Computational Linguistics

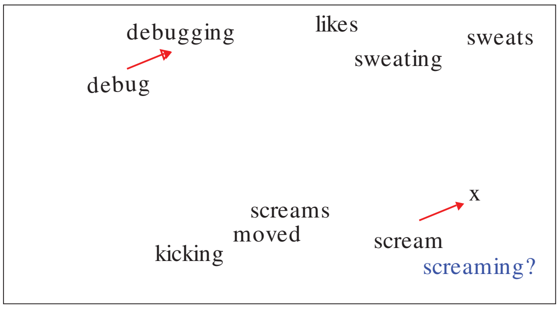

Figure 2: When $a^* - a$ is small and b and b^* are close, the expected answer may be returned even when the offsets are inconsistent (here *screaming* is closest to x).

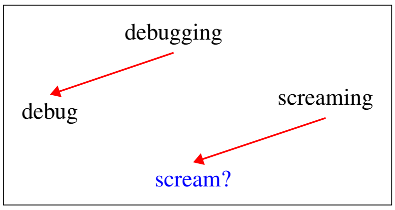

Figure 3: Reversing the direction of the task.

methods. First, we propose new baselines that perform the task without using the offset $a^* - a$ and argue that the performance of the offset method should be compared to those baselines. Second, we measure how the performance of the method is affected by reversing the direction of each analogy problem (Figure 3). If the method truly measures offset consistency, this reversal should not affect its accuracy.

2 Analogy functions

We experiment with the following functions. In all of the methods, every word in the vocabulary can serve as a guess, except when a, a^* or b are explicitly excluded as noted below. Since the size of the vocabulary is typically very large, chance performance, or the probability of a random word in the vocabulary being the correct guess, is extremely low.

VANILLA: This function implements the offset method literally (Equation 2).

ADD: The x^* obtained from Equation 2 is often trivial (typically equal to b). In practice, most studies exclude a, a^* and b from consideration:

$$x^* = \underset{x' \notin \{a, a^*, b\}}{\text{argmax}} \cos(x', a^* - a + b) \tag{4}$$

ONLY-B: This method ignores both a and a^* and simply returns the nearest neighbor of b:

$$x^* = \underset{x' \notin \{a, a^*, b\}}{\text{argmax}} \cos(x', b) \tag{5}$$

As shown in Figure 2, this baseline is likely to give a correct answer in cases where $a^* - a$ is small and b^* happens to be the nearest neighbor of b.

IGNORE-A: This baseline ignores a and returns the word that is most similar to both a^* and b:

$$x^* = \underset{x' \notin \{a, a^*, b\}}{\text{argmax}} \cos(x', a^* + b) \tag{6}$$

A correct answer using this method indicates that b^* is closest to a point y that lies mid-way between a^* and b (i.e. that maximizes the similarity to both words).

ADD-OPPOSITE: This function takes the logic behind the ONLY-B baseline a step further – if the neighborhood of b is sufficiently sparse, we will get the correct answer even if we go in the *opposite* direction from the offset $a^* - a$:

$$x^* = \underset{x' \notin \{a, a^*, b\}}{\text{argmax}} \cos(x', -(a^* - a) + b) \tag{7}$$

MULTIPLY: Levy and Goldberg (2014) show that Equation 2 is equivalent to adding and subtracting cosine similarities, and propose replacing it with multiplication and division of similarities:

$$x^* = \underset{x' \notin \{a, a^*, b\}}{\text{argmax}} \frac{\cos(x', a^*) \cos(x', b)}{\cos(x', a)} \tag{8}$$

REVERSE (ADD): This is simply ADD applied to the reverse analogy problem: if the original problem is *debug : debugging :: scream :* ___, the reverse problem is *debugging : debug :: screaming :* ___. A substantial difference in accuracy between the two directions in a particular type of analogy problem (e.g., base-to-gerund compared to gerund-to-base) would indicate that the neighborhoods of one of the word categories (e.g., gerund) tend to be sparser than the neighborhoods of the other type (e.g., base).

REVERSE (ONLY-B): This baseline is equivalent to ONLY-B, but applied to the reverse problem: it returns b^*, in the notation of the original analogy problem.

	a	a^*	n
Common capitals:	*athens*	*greece*	506
All capitals:	*abuja*	*nigeria*	4524
US cities:	*chicago*	*illinois*	2467
Currencies:	*algeria*	*dinar*	866
Nationalities:	*albania*	*albanian*	1599
Gender:	*boy*	*girl*	506
Plurals:	*banana*	*bananas*	1332
Base to gerund:	*code*	*coding*	1056
Gerund to past:	*dancing*	*danced*	1560
Base to third person:	*decrease*	*decreases*	870
Adj. to adverb:	*amazing*	*amazingly*	992
Adj. to comparative:	*bad*	*worse*	1332
Adj. to superlative:	*bad*	*worst*	1122
Adj. un- prefixation:	*acceptable*	*unacceptable*	812

Table 1: The analogy categories of Mikolov et al. (2013a) and the number of problems per category.

3 Experimental setup

Analogy problems: We use the analogy dataset proposed by Mikolov et al. (2013a). This dataset, which has become a standard VSM evaluation set (Baroni et al., 2014; Faruqui et al., 2015; Schnabel et al., 2015; Zhai et al., 2016), contains 14 categories; see Table 1 for a full list. A number of these categories, sometimes referred to as "syntactic", test whether the structure of the space captures simple morphological relations, such as the relation between the base and gerund form of a verb (*scream* : *screaming*). Others evaluate the knowledge that the space encodes about the world, e.g., the relation between a country and its currency (*latvia* : *lats*). A final category that doesn't fit neatly into either of those groups is the relation between masculine and feminine versions of the same concept (*groom* : *bride*). We follow Levy and Goldberg (2014) in calculating separate accuracy measures for each category.

Semantic spaces: In addition to comparing the performance of the analogy functions within a single VSM, we seek to understand to what extent this performance can differ across VSMs. To this end, we selected three VSMs out of the set of spaces evaluated by Linzen et al. (2016). All three spaces were produced by the skip-gram with negative sampling algorithm implemented in word2vec (Mikolov et al., 2013b), and were trained on the concatenation of ukWaC (Baroni et al., 2009) and a 2013 dump of the English Wikipedia.

The spaces, which we refer to as s_2, s_5 and s_{10}, differed only in their context window parameters. In s_2, the window consisted of two words on either side of the focus word. In s_5 it included five words on either side of the focus word, and was "dynamic" – that is, it was expanded if any of the context words were excluded for low or high frequency (for details, see Levy et al. (2015)). Finally, the context in s_{10} was a dynamic window of ten words on either side. All other hyperparameters were set to standard values.

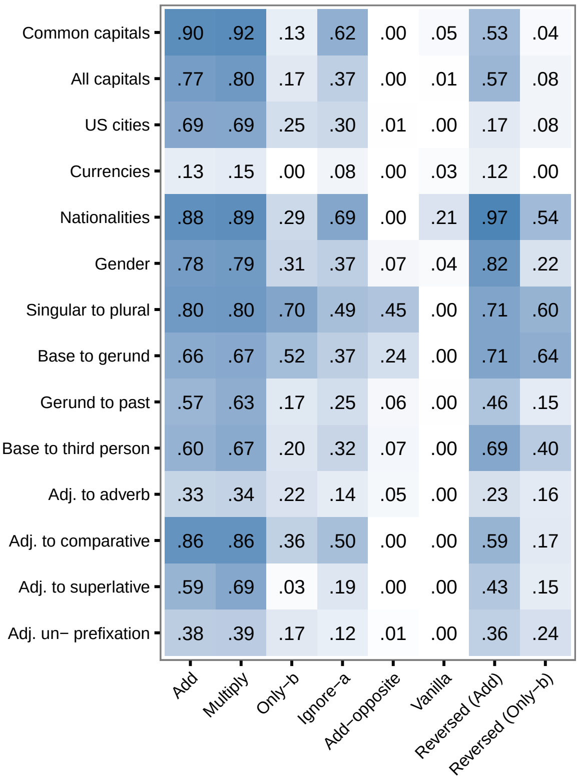

Figure 4: Accuracy of all functions on space s_5.

4 Results

Baselines: Figure 4 shows the success of all of the analogy functions in recovering the intended analogy target b^* in space s_5. In line with Levy and Goldberg (2014), there was a slight advantage for MULTIPLY over ADD (mean difference in accuracy: .03), as well as dramatic variability across categories (ranging from .13 to .90 in ADD). This variability cuts across the distinction between the world-knowledge and morphological categories; performance on currencies and adjectives-to-adverbs was poor, while performance on capitals and comparatives was high.

Although ADD and MULTIPLY always outperformed the baselines, the margin varied widely across categories. The most striking case is the plurals category, where the accuracy of ONLY-B reached .70, and even ADD-OPPOSITE achieved

Space	ADD	ADD - IGNORE-A	ADD - ONLY-B
s_2	.53	.41	.42
s_5	.6	.29	.36
s_{10}	.58	.26	.33

Table 2: Overall scores and the advantage of ADD over two of the baselines across spaces.

a decent accuracy (.45). Taking a^* but not a into account (IGNORE-A) outperformed ONLY-B in ten out of 14 categories. Finally, the poor performance of VANILLA confirms that a, a^* and b must be excluded from the pool of potential answers for the offset method to work. When these words were not excluded, the nearest neighbor of $a^* - a + b$ was b in 93% of the cases and a^* in 5% of the cases (it was never a).

Reversed analogies: Accuracy decreased in most categories when the direction of the analogy was reversed (mean difference -0.11). The changes in the accuracy of ADD between the original and reversed problems were correlated across categories with the changes in the performance of the ONLY-B baseline before and after reversal (Pearson's $r = .72$). The fact that the performance of the baseline that ignores the offset was a reliable predictor of the performance of the offset method again suggests that the offset method when applied to the Mikolov et al. (2013a) sets jointly evaluates the consistency of the offsets and the probability that b^* is the nearest neighbor of b.

The most dramatic decrease was in the US cities category (.69 to .17). This is plausibly due to the fact that the city-to-state relation is a many-to-one mapping; as such, the offsets derived from two specific city-states pairs — e.g., *Sacramento:California* and *Chicago:Illinois* — are unlikely to be exactly the same. Another sharp decrease was observed in the common capitals category (.9 to .53), even though that category is presumably a one-to-one mapping.

Comparison across spaces: The overall accuracy of ADD was similar across spaces, with a small advantage for s_5 (Table 2). Yet the breakdown of the results by category (Figure 5) shows that the similarity in average performance across the spaces obscures differences across categories: s_2 performed much better than s_{10} in some of the morphological inflection categories (e.g., .7 compared to .44 for the base-to-third-person relation), whereas s_{10} had a large advantage in some of the world-knowledge categories (e.g., .68 compared to .42 in the US cities category). The advantage of smaller window sizes in capturing "syntactic" information is consistent with previous studies (Redington et al., 1998; Sahlgren, 2006). Note also that overall accuracy figures are potentially misleading in light of the considerable variability in the number of analogies in each category (see Table 1): the "all capitals" category has a much greater effect on overall accuracy than gender, for example.

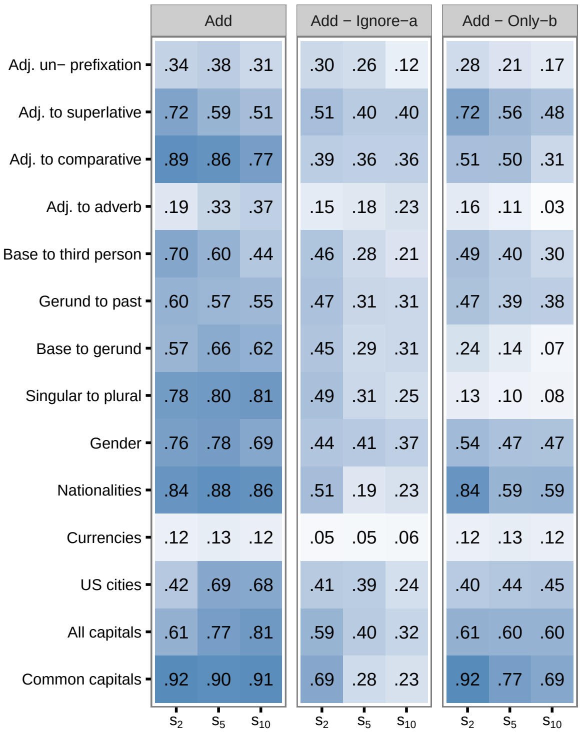

Figure 5: Comparison across spaces. The leftmost panel shows the accuracy of ADD, and the next two panels show the improvement in accuracy of ADD over the baselines.

Spaces also differed in how much ADD improved over the baselines. The overall advantage over the baselines was highest for s_2 and lowest for s_{10} (Table 2). In particular, although accuracy was similar across spaces in the nationalities and common capitals categories, much more of this accuracy was already captured by the IGNORE-A baseline in s_{10} than in s_2 (Figure 5)

5 Discussion

The success of the offset method in solving word analogy problems has been taken to indicate that systematic relations between words are represented in the space as consistent vector offsets

(Mikolov et al., 2013c). The present note has examined potential difficulties with this interpretation. A literal ("vanilla") implementation of the method failed to perform the task: the nearest neighbor of $a^* - a + b$ was almost always b or a^*.[1] Even when those candidates were excluded, some of the success of the method on the analogy sets that we considered could also be obtained by baselines that ignored a or even both a and a^*. Finally, reversing the direction of the analogy affected accuracy substantially, even though the same offset was involved in both directions.

The performance of the baselines varied widely across analogy categories. Baseline performance was poor in the adjective-to-superlative relation, and was very high in the plurals category (even when both a and a^* were ignored). This suggests that analogy problems in the plural category category may not measure whether the space encodes the single-to-plural relation as a vector offset, but rather whether the plural form of a noun tends to be close in the vector space to its singular form. Baseline performance varied across spaces as well; in fact, the space with the weakest overall performance (s_2) showed the largest increases over the baselines, and therefore the most evidence for consistent offsets.

We suggest that future studies employing the analogy task report the performance of the simple baselines we have suggested, in particular ONLY-B and possibly also IGNORE-A. Other methods for evaluating the consistency of vector offsets may be less vulnerable to trivial responses and neighborhood structure, and should be considered instead of the offset method (Dunbar et al., 2015).

Our results also highlight the difficulty in comparing spaces based on accuracy measures averaged across heterogeneous and unbalanced analogy sets (Gladkova et al., 2016). Spaces with similar overall accuracy can vary in their success on particular categories of analogies; effective representations of "world-knowledge" information are likely to be useful for different downstream tasks than effective representations of formal linguistic properties. Greater attention to the fine-grained strengths of particular spaces may lead to the development of new spaces that combine these strengths.

Acknowledgments

I thank Ewan Dunbar, Emmanuel Dupoux, Omer Levy and Benjamin Spector for comments and discussion. 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

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 238–247.

Ewan Dunbar, Gabriel Synnaeve, and Emmanuel Dupoux. 2015. Quantitative methods for comparing featural representations. In *Proceedings of the 18th International Congress of Phonetic Sciences*.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, Colorado, May–June. Association for Computational Linguistics.

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pages 8–15, San Diego, California, June. Association for Computational Linguistics.

Omer Levy and Yoav Goldberg. 2014. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the Eighteenth Conference on Computational Language Learning*, pages 171–180.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

[1] A human with any reasonable understanding of the analogy task is likely to also exclude a, a^* and b as possible responses, of course. However, such heuristics that are baked into an analogy solver, while likely to improve its performance, call into question the interpretation of the success of the analogy solver as evidence for the geometric organization of the underlying semantic space.

Tal Linzen, Emmanuel Dupoux, and Benjamin Spector. 2016. Quantificational features in distributional word representations. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics (*SEM 2016)*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *Proceedings of ICLR*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013c. Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT*, pages 746–751.

Martin Redington, Nick Chater, and Steven Finch. 1998. Distributional information: A powerful cue for acquiring syntactic categories. *Cognitive Science*, 22(4):425–469.

Magnus Sahlgren. 2006. *The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. Ph.D. thesis, Stockholm University.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of EMNLP*.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Peter D Turney. 2012. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research*, pages 533–585.

Michael Zhai, Johnny Tan, and Jinho D Choi. 2016. Intrinsic and extrinsic evaluations of word embeddings. In *Thirtieth AAAI Conference on Artificial Intelligence*.