Evaluating the Consistency of Word Embeddings from Small Data

Jelke Bloem[♠] **Antske Fokkens**[♣] **Aurélie Herbelot**[♢]
♠ Institute for Logic, Language and Computation. University of Amsterdam
♣ Computational Lexicology and Terminology Lab. Vrije Universiteit Amsterdam
♢ Center for Mind/Brain Sciences / DISI, University of Trento
j.bloem@uva.nl, antske.fokkens@vu.nl, aurelie.herbelot@unitn.it

Abstract

In this work, we address the evaluation of distributional semantic models trained on smaller, domain-specific texts, particularly philosophical text. Specifically, we inspect the behaviour of models using a pre-trained background space in learning. We propose a measure of *consistency* which can be used as an evaluation metric when no in-domain gold-standard data is available. This measure simply computes the ability of a model to learn similar embeddings from different parts of some homogeneous data. We show that in spite of being a simple evaluation, consistency actually depends on various combinations of factors, including the nature of the data itself, the model used to train the semantic space, and the frequency of the learned terms, both in the background space and in the in-domain data of interest.

1 Introduction

Distributional semantic (DS) models (Turney and Pantel, 2010; Erk, 2012; Clark, 2015) typically require very large corpora to construct accurate meaning representations of words (Bengio et al., 2003). This big data methodology presents challenges when working with text in a specific domain or a low-resource language. In this paper, we are interested in modeling concepts in philosophical corpora, which are far smaller than a typical web corpus. Instead of training directly on the philosophical in-domain data, which is too sparse for learning, we rely on a pre-trained background semantic space, thus simulating a speaker with some linguistic knowledge coming to a new domain.

Our focus is the evaluation problem encountered when working with domain-specific data. DS models are typically evaluated on gold standard datasets containing word association scores elicited from human subjects (e.g. Bruni et al., 2014; Hill et al., 2015). Beside the limited practical use of such evaluation metrics (e.g. Gladkova and Drozd, 2016), this is not a feasible method for evaluating DS models in low-resource situations. When domain-specific terminology is used and the meaning of words possibly deviate from their most dominant sense, creating regular evaluation resources can require significant time investment from domain experts. Evaluation metrics that do not depend on such resources are valuable. Thus, we introduce the metric of **consistency**, which requires a model to learn similar word embeddings for a given term across similar sources, for example, two halves of a book.

Philosophical texts make a suitable case study for out-of-domain data, as words may have very different meanings in philosophy than in general usage. For example, while a *proposition* is synonymous for *offer* or *proposal* in ordinary language, in philosophy it is, among other things, a bearer of truth-value (McGrath and Frank, 2018). Furthermore, philosophical writing is often precise and terminology tends to be defined or at least discussed in the text, so there should be enough information for modeling meaning even when working with small data, for instance in one or multiple works by a particular philosopher or from a particular philosophical tradition. Last but not least, the field of philosophy could benefit from this type of modeling — although philosophers have not yet made broad use of computational methods (Betti et al., 2019), it has been shown that new insights can be obtained using an information retrieval tool based on a distributional semantic model of digitalized philosophical texts (Ginammi et al., in press).

Using philosophical data, we perform a battery of tests which reveal interesting properties of con-

Proceedings of Recent Advances in Natural Language Processing, pages 132–141,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_016

sistency. We show that in spite of being a simple evaluation, consistency actually depends on various combinations of factors, including the nature of the data itself, the model used to train the semantic space, and the frequency of the learned terms, both in the background space and in the in-domain data of interest. This leads us to conclude that the evaluation of in-domain word embeddings from small data has to be controlled extremely carefully in order not to draw incorrect conclusions from experimental results.

2 Related Work

Learning embeddings for rare words is a very challenging process (Luong et al., 2013). Word2Vec (W2V, Mikolov et al., 2013a)'s skipgram model can learn embeddings from tiny data after modification, as shown by Herbelot and Baroni (2017) when it consists of just a single highly informative definitional sentence. However, philosophical data is typically *small data* rather than tiny data. While tiny data consists of a single definitional sentence, our small data consists of multiple context sentences per term that are not necessarily definitional. Herbelot and Baroni's (2017) Nonce2Vec (N2V) has not been tested on this type of data. W2V has been tested on smaller datasets, but was found to be suboptimal (Asr et al., 2016) and surpassed by SVD models on a 1 million word dataset (Sahlgren and Lenci, 2016).

Different DS evaluations test different aspects of the learned embeddings (i.e. Wang et al., 2019). Most existing methods are however not easily applicable to our task. The typical evaluation of comparing embedding similarities to a gold standard of word similarity scores, such as the SimLex-999 dataset (Hill et al., 2015) cannot be applied, because we are interested in the representation of specific terms: even if these terms are present in the evaluation set, their meaning in the philosophical domain is likely to differ. Manually creating a domain-specific resource requires labor-intensive effort by domain experts, which makes it impractical to port standard datasets to a specific type of corpora. This holds also for other evaluation methods such as analogy scores (Mikolov et al., 2013b), as well as coherence (Schnabel et al., 2015), which is based on the idea that pairs of similar words should be close in semantic space.

Methods where human raters directly respond to output of the model, such as comparative intrinsic evaluation (Schnabel et al., 2015) are interesting, but require domain experts, as well as instructions that elicit the desired type of semantic relation (i.e. similarity). Extrinsic evaluation requires a downstream task that can be evaluated, but in this use case we are interested in the information encoded by the DS model itself. QVEC (Tsvetkov et al., 2015) evaluates by aligning dimensions of a semantic space to linguistic features, but we are interested only in evaluating some vectors rather than an entire space (target term vectors but not the background space vectors), and this approach requires language-specific resources.

Nooralahzadeh et al. (2018) evaluate domain-specific embeddings by building a query inventory for their domain from a glossary containing synonym, antonym and alternative form information. Unfortunately, such structured glossaries are generally not available for specific philosophers. Hellrich and Hahn (2016) test their models for reliability in a study investigating historical English and German texts, another relatively low-resource domain. Their reliability metric involves training three identically parametrized models, and comparing the nearest neighbors of each word in each model using a modified Jaccard coefficient. This metric does not require any language-specific data, but it mainly serves as a test of the impact of the sources of randomness in Word2Vec, and not as a measure of the systematic semantic differences across various data sources.

3 Consistency Metric

We propose **consistency** as a useful metric to evaluate word embeddings in the absence of domain-specific evaluation datasets. We consider a model to be consistent if its output does not vary when its input should not trigger variation (i.e. because it is sampled from the same text). Thus, a model can only be as consistent as the input data it is trained on and it requires the experimenter to compute data consistency in addition to vector space consistency.

To evaluate *data consistency*, we create vectors for target terms in a domain corpus under two conditions: a) random sampling; b) equal split. The 'equal split' condition simply corresponds to splitting the data in the middle, thus obtaining two subcorpora of equal size and in diachronic order. Given a pre-trained background space kept frozen across experiments, the vector representation of a target is generated by simple vector addition over

its context words. Therefore, the obtained vector directly represents the context the target term occurs in, and consequently, similar representations (in terms of cosine similarity) mean that the target term is used in a similar way in different parts of a book/corpus, and is thus consistently learnable. Crucially, though, this measure may interact with data size. Kabbach et al. (2019) recently noted a *sum effect* in the additive model, where summed vectors are close to each other. It may be the case that additive model vectors summed over more context data contain more information and may have higher similarity between each other, resulting in higher consistency scores. We test this in Section 6. When randomly sampling, we limit the number of sentences per sample to control for this.

To evaluate *space consistency*, we create identically parametrized models as in Hellrich and Hahn's (2016) reliability metric, but over different parts of the data, with the data being split in the middle, as just described. We consider two ways of comparing two vectors $\vec{a_1}$ and $\vec{a_2}$: by similarity, where a higher cosine similarity indicates more consistency, or by nearest neighbor rank, where a higher rank of $\vec{a_1}$ among the nearest neighbors of $\vec{a_2}$ indicates more consistency. Every vector in the background space, as well as $\vec{a_2}$, is ranked by cosine similarity to $\vec{a_1}$ to compute this rank value.

Although it is more complex than having a single metric, we must consider both rank and similarity simultaneously: rank is a more relative metric and helps to ground the similarity value in the local context of the target term. A vector with 0.8 similarity but lower rank is a worse result than a vector with 0.8 similarity and a high rank, as the low rank means that the vectors are in a dense part of the semantic space and a very high similarity is required to consistently identify which of the neighbouring vectors refers to the same concept. Conversely, a low-similarity, high-rank vector can be a cause for scepticism, as it may have been placed far out from the rest of the semantic space.

We take consistency to be a desirable property of word embeddings at the level of a certain domain. Of course, consistency only measures one specific desirable property of embeddings and should thus not be interpreted as a general quality or accuracy score. But even taken on its own, we will show that it exhibits complex behavior with respect to data, background vectors and term frequency.

4 Task Description

Our overall aim is to obtain consistent embeddings of terms central to the works of Willard Van Orman Quine (1908-2000), an influential 20th century philosopher and logician. As the meaning of terms may differ between authors and even between books by the same author, we need to learn such embeddings from small data, bounded by the occurrences of the term in one particular book.

Quine datasets We build novel datasets based on a corpus of 228 philosophical articles, books and bundles written by Quine, with a focus om two of Quine's books: *A System of Logistic* (Quine, 1934) and *Word & Object* (Quine, 1960). These Quine texts are part of a larger corpus of philosophical texts, which is still being compiled, that are central to the history of scientific ideas (Betti and van den Berg, 2016). We focus on these particular works from the corpus because testing consistency is best done on homogeneous data, and our philosophy domain experts informed us that Quine was a remarkably stable philosopher in his outlook (Betti and Oortwijn, p.c.).

The first book is a formula-heavy logic book, deviating strongly from ordinary language. Such a technical book is particularly likely to be internally consistent. It contains 80,279 tokens after tokenization and manual replacement of formulas with special tokens. The second book is more textual and consists of standard philosophical argumentation. Our domain experts consider it conceptually consistent. It contains 133,240 tokens after tokenization. The full corpus of the 228 Quine articles contains 1.7 million tokens and is pre-processed with NLTK (Bird et al., 2009) for sentence splitting and tokenization. A less common preprocessing step we took was to remove one-character tokens from the texts. These works contain many one-letter variable names, logical symbols and other formal language that a model might otherwise use to position vectors of Quine terminology in particular areas of the semantic space, as these tokens are highly infrequent in the general domain.

To obtain terms that would be relevant to model, we automatically extract terms from the two books' indexes, as the most important terminology is likely to be listed there. We include multi-word terms, and divide the terms into 70%/30% subsets for training and testing, resulting in a 157 / 67 split for *Logistic* and a 184 / 79 split for *Word & Ob-*

ject. The target terms thus differ per book, as each book lists different terms in its index. Instead of this automatic approach to obtaining target terms, an expert-created resource could provide a better set of target terms, if available. If neither this nor a terms glossary or index of terms is available, keyword extraction methods could be used as an alternative way of obtaining terms for evaluation. In cases where the model will not be used for any analysis of domain-specific content downstream, it may be sufficient to randomly select words from the text as target terms.

Next, we derive datasets from this corpus using our two conditions for data consistency: random sampling and equal split. In *random sampling*, for each target term that meets a frequency cutoff of 10, we randomly select five non-overlapping samples of up to 10 random sentences that contain the target term, divided evenly across the samples if the term occurs in fewer than 50 sentences. This gives us the datasets *Quine-WordObject-rnd* (with Word & Object core terms as target terms), *Quine-Logistic-rnd* (with System of Logistic core terms) for our two books of interest, and *Quine-all-rnd* sampled from the full Quine corpus, where we also use the Word & Object core terms as target terms.[1] In the *equal split* condition, we divide a book into two halves at a chapter boundary, and extract all sentences containing index terms that meet a frequency cuf-off of 2 in each half, resulting in the datasets *Quine-WordObject* and *Quine-Logistic*. With random sampling, we intend to capture the basic consistency of the model. With equal split, we aim to capture consistency across potential meaning development throughout the book.[2]

Wikipedia dataset For cross-domain comparison, we apply our method to a 140M word preprocessed Wikipedia snapshot using the same random sampling process. As target terms, we used 300 randomly sampled one-word Wikipedia page titles, following Herbelot and Baroni (2017).

5 Method

Before evaluating whether we have *space consistency*, we must establish to what extent we have *data consistency*, following our argumentation in Section 3. To obtain an embedding for a new target term, we use an additive model over its context words, using as background space ordinary language representations. For the in-domain context, we use a window size of 15, with the window being restricted to the sentence. The background space is based on a Wikipedia snapshot of $1.6B$ words trained with Word2Vec's Gensim implementation with default parameters, and containing 259,376 word vectors in 100 dimensions. For each target term, context words undergo subsampling, which randomly drops higher-frequency words.[3] The vectors of the remaining context words are summed to create a vector for the target term. This additive model was used by Lazaridou et al. (2017) for their textual data, and was shown by Herbelot and Baroni (2017) to work reasonably well on tiny data. We calculate the vectors separately per sample (or book half), yielding comparable term vectors.

Next, we turn to *space consistency*. We use our consistency metric to evaluate two models that are suited to learning embeddings from small data: Nonce2Vec (Herbelot and Baroni, 2017) and an SVD-reduced count-based model over concatenations of our datasets with general-domain data.

The first model, Nonce2Vec, modifies W2V's 'skip-gram' model (Mikolov et al., 2013a) in a way that is inspired by *fast mapping* (Carey and Bartlett, 1978) in humans. Human learners can acquire new words from just a single token and this process of fast mapping appears to build on concepts that are already known (Coutanche and Thompson-Schill, 2014). Nonce2Vec models this through incremental learning, an initial high learning rate, greedy processing, and parameter decay. To simulate the existence of background knowledge, Nonce2Vec maps its novel word vectors into a previously learned semantic space, based on the aforementioned Wikipedia snapshot and the same subsampling procedure. Target term vectors are initialized to their sum vector from the additive model. For each sentence, the model is trained on the target term, only updating the weights for that term and freezing all other network parameters. The learning rate and context window size decay in proportion to the number of times the target term has been seen, and the subsampling rate increases per sentence.

Secondly, we try a count-based approach, creat-

[1] Word & Object touches upon much of Quine's work, so its terminology can be considered representative.

[2] While our datasets are derived from copyrighted works and cannot be shared, we provide replication instructions, term lists and code here: https://bloemj.github.io/quine2vec/

[3] Some promising alternative subsampling methods for tiny data were recently discussed by Kabbach et al. (2019).

Dataset	cos-sim	rank
Quine-WordObject	0.938	1
Quine-Logistic	0.907	22.4
Quine-WordObject-rnd	0.919	1
Quine-Logistic-rnd	0.935	1
Quine-all-rnd	0.953	1
Wiki-rnd	0.927	1.001

Table 1: Consistency metrics on different data sets using the additive model.

n	cos-sim
1	0.794
2	0.837
3	0.905
4	0.923
8	0.956
all	0.987

Table 2: Data consistency for the term *analytical hypotheses* in Word & Object when varying the number of sentences per sample n.

ing vectors over the general-domain and in-domain data at the same time. In this procedure, we concatenate a particular Quine dataset with a 140M word Wikipedia corpus sample, in which the Quine terms are marked with special tokens in order to be trained separately from the same term in the Wikipedia data. We create embeddings from this corpus, applying PPMI weighting and singular value decomposition (SVD) to reduce the models to 100 dimensions, to match the dimensionality of our other models and because factorized count models have been shown to work well on smaller datasets (Sahlgren and Lenci, 2016). We use the *Hyperwords* implementation of Levy et al. (2015), with a window size of 5, and other hyperparameters set to the default values.

In both the above approaches, we can then compute vector space consistency between different vectors learned for the same term over different splits of the data.

6 Consistency of Data

We start by applying the additive model to quantify data consistency on the different datasets described in Section 4. We compute average similarities and nearest neighbor ranks over the vectors of all target terms in a dataset. For the randomly sampled data sets, we have five vectors per term, one from each sample, and compute the metrics over all unique combinations of 2 vectors. For the equal split setting, we compare the term vectors summed over each half of the book.

The additive model produces highly consistent embeddings on the training data: for most terms, the vectors summed over each book half are each other's nearest neighbors in the background space. This trend is also observed for the test sets presented in Table 1, where we observe high consistency for the embeddings from both books.

Using the book halves of System of Logistic (*Quine-Logistic*) gives us a slightly lower data consistency score than random sampling from that book (*Quine-Logistic-rnd*), possibly because the meaning of a term may evolve from the first half to the second half of a book. This suggests some utility of the data consistency measure in quantifying meaning development throughout a text, as long as other factors are controlled for. We also see that the Wikipedia domain data (*Wiki-rnd*) is less consistent than the Quine domain data (*Quine-all-rnd*), which is to be expected as it contains more diverse text.

These results seem to indicate that the additive model provides consistent embeddings. This means that it must be possible to learn consistent embeddings from these datasets, at least up to the consistency values reported here, as the additive model directly represents the contexts that predictive models use for training.

As already mentioned, however, the factor of data size may interfere with consistency. We do observe in Table 1 that the consistency of data sampled across the full Quine corpus is higher. Although we limited our samples to 10 sentences per term, not every core Quine term is used frequently enough to have 5 samples with the maximum size of 10 sentences. Specifically, in the full Quine dataset, 68.6% of terms reach the maximum size, while in the Word & Object dataset, only 32.1% of terms reach it. In the Wiki set, this is 90.9%, showing that its lower consistency is not explained by limited data. To fully control for data size, we would need to use artificial data: if we control for the number of sentences, the number of words and the number of words subsampled still affect data size. As we are mainly interested in the quality of models on our own philosophical corpus, we leave this for future work.

Instead, we test the effect of data size by summing two vectors for the same term over varying numbers of sentences, and computing the consis-

tency between them. Table 2 shows a clear effect of data size: vectors summed over more sentences have higher data consistency. This shows that data consistency should ideally be computed within the constraints of a particular data size, because vectors summed over more context are more informative and thus more consistent.

7 Consistent Spaces

Having established that our data is consistent even with fairly small samples, we proceed to use two small data approaches to place terms consistently in vector space. We start with Nonce2Vec (N2V), which uses the sum vectors from the additive model for initialization and trains only on that vector, as it does not update the vectors in the background space, only that of the target term.

For this experiment, we modified N2V in two ways. Firstly, it now takes multiple sets of input sentences per target term, one from each sample or book half, and trains each term on all sets separately, resulting in multiple term vectors, one over each sample. Secondly, we implemented the consistency metrics described in Section 3 for comparing the different sample vectors and analyzing their position in the background space.

Using N2V's default parameters, we obtain low consistency scores. While N2V was designed for learning from a dataset with one sentence per term, the terms in our dataset occur in more sentences. A likely consequence of this difference, having small data instead of tiny data, is that the default parameters may include a too high learning rate and N2V's parameter decay may be too fast. A learning rate that is too high can result in models with low stability. To adapt to small data, we tune N2V's parameters on the full Quine dataset with the training set of target terms. We performed a grid search following a parameter space containing different learning rates ($[0.1, 0.5, \mathbf{1}, 1.5]$), the number of negative samples ($[1, \mathbf{3}, 5]$), the subsampling rate ($[100, 3000, 5000, \mathbf{10000}, 20000]$), learning rate decay ($[30, \mathbf{70}, 100, 180]$), subsampling rate decay ($[1.0, 1.3, \mathbf{1.9}, 2.5]$) window decay ($[1, 3, \mathbf{5}]$), window size ($[\mathbf{15}]$). Bold values are the best performing values in Herbelot and Baroni (2017) or defaults of N2V. We obtain our best performance with a learning rate of 0.1, 5 negative samples, a learning rate decay of 30 and a subsampling decay factor of 2.5.

We obtain fairly consistent embeddings with these parameters on the test set, as shown in Table 3: the vectors learned from the two book halves in the *Quine-WordObject* and *Quine-Logistic* datasets are often each other's nearest neighbour, with average nearest neighbour ranks of 1.21 and 1.48, respectively. Surprisingly, although this model is initialized using Wikipedia background vectors, that domain (*Wiki-rnd*) fares the worst in terms of consistency, as it does in the additive model. In general, these vector space consistency scores are lower than the data consistency scores we saw before, so there is room for improvement.

Dataset	cos-sim	rank
Quine-WordObject	0.686	1.21
Quine-Logistic	0.748	1.48
Quine-WordObject-rnd	0.695	2.11
Quine-Logistic-rnd	0.743	1
Quine-all-rnd	0.717	1.59
Wiki-rnd	0.589	507.8

Table 3: Consistency metrics on different data sets for the Nonce2Vec-based models.

Dataset	cos-sim
Quine-WordObject-rnd	0.352
Quine-Logistic-rnd	0.436
Quine-all-rnd	0.440
Wiki-rnd	0.321

Table 4: Consistency on different data sets for the SVD models.

We therefore turn to our other approach that is not based on the additive model: the SVD models over the concatenation of in-domain and general-domain data. When concatenating the datasets, we have to ensure that the target terms in our random samples of in-domain data are trained separately from the same term in the general domain and in other samples. We therefore mark them with a different ID for each sample. As before, we compute cosine similarities between these target terms from different samples to measure consistency.

Table 4 shows that the resulting embeddings are not very consistent, with much lower average cosine similarities between the samples that does not reflect the consistency of the data, as indicated by the additive model in Table 1. The consistency of the SVD vectors is also lower than that of the Nonce2Vec vectors from the previous experiment.

One possible explanation for the difficulty that both of these models have in learning from our data is in the bridging of the domain gap between the

Group of terms	similarity
System of Logistic	0.323
Word & Object	0.366
Q-High-freq W-Low-freq	0.735
Q-Low-freq W-Low-freq	0.417
Q-Low-freq W-High-freq	0.109
Q-High-freq W-High-freq	0.078

Table 5: Average similarities between Quine vectors and Wiki vectors in our SVD model. Q = Quine, W = Wiki.

Wikipedia general-domain spaces and the Quine terminology. To quantify the difference between domains, we selected all sentences from the Quine corpus containing 35 target terms and concatenated them with our 140M word Wikipedia sample, as in the previous experiment. These terms were selected to be either high-frequent or low-frequent in the Quine domain and either high-frequent or low-frequent in the general domain. Again, the Quine terms were marked in order to be trained separately from the same term in the Wikipedia domain, and we created a SVD model. In this SVD model, we computed the cosine similarities between each Quine term and its Wikipedia counterpart, and take this to be a measure of domain difference.

Table 5 shows a clear effect of term frequency. We grouped all terms according to two factors: their frequency in the Quine book they were selected for (low, relative frequency[4] < 0.0005 or high, relative frequency > 0.001) and their frequency in the Wikipedia domain (low, RF < 0.000025 or high, RF > 0.00005).[5] We observe that infrequent terms with a dominant philosophical sense such as *stimulus* have more similar vectors in both domains despite their sparsity in both corpora. Generally, terms that are highly frequent in the Quine-domain but have low frequency in the Wikipedia domain are more similar between the two domains (*Q-High-freq W-Low-freq*). To a lesser extent, this is also true for terms that are low-frequent in both domains.

This result indicates that bridging the domain gap should be easier with these philosophical core terms than with frequent Wikipedia terms. The fact that our models are less consistent on Wikipedia data also indicates that the generality of this domain is more relevant than any specific differences with the Quine domain. It must therefore be possible to learn good representations from this data by using background knowledge from the Wikipedia domain, but the models we tested did not reach the level of consistency of the additive model.

[4] $\frac{F}{C}$ where F is the term frequency and C is the corpus size.
[5] Different thresholds are necessary for the larger corpus.

Dataset	cos-sim	rank
Quine-WordObject-rnd	0.352	22,947
Quine-Logistic-rnd	0.353	24,513
Quine-all-rnd	0.382	17,262
Wiki-rnd	0.475	2,902

Table 6: Average similarities between learned in-domain term vectors and pretrained general-domain background vector on different data sets for the Nonce2Vec-based models.

For better or for worse, our models do move away from what is in the background space. In our Nonce2Vec experiment on the *Quine-all-rnd* dataset, we also measured the average cosine similarity and nearest neighbour rank of the pretrained Word2Vec term vector from the background space, compared to the vectors we learned for that same term from the in-domain data. These numbers, shown in Table 6, reveal that the model does not stay close to the pre-trained background vectors in order to achieve high consistency, which could be a risk if consistency was used as a learning signal in combination with an invariant initialization. Furthermore, the vectors learned from the Wiki data are closer to the pre-trained vectors than those learned from the Quine data. This is expected of a good model, as there is no domain gap to bridge when training with Wikipedia context sentences into a Wikipedia background space. This also means that the vector representations for terms as used by Quine become more distinct after training, as our philosophy domain experts would expect of a good meaning representation of these in-domain terms.

We must again note that consistency is not the only desirable property of word embeddings. Unfortunately, other properties are more difficult to evaluate on low-resource data. Without a domain-specific evaluation set, we can only explore issues with quality by examining nearest neighbors of vectors that our metric marks as perfectly consistent. We observe both in our results, illustrated by cherry-picked examples from the Nonce2Vec model on the *Quine-WordObject* dataset. Table 7 shows that the nearest neighbours for both book half vectors for the term *talking* (*Word & Object*) look bad. The vectors' nearest neighbours are some

Term		$\vec{a_1}$ NNs	$\vec{a_2}$ NNs
talking	1	wrongfulness	axiomatically
	2	axiomatically	epiphenomenon
	3	particularized	impredicative
verbs	1	logophoric	logophoric
	2	deverbal	resumptive
	3	adpositions	countability
	4	uninflected	adverbials

Table 7: Qualitative examination of some nearest neighbours of target term vectors computed over book halves 1 and 2 of *Word & Object*.

apparently unrelated words yet they are closest to each other (similarity 0.751). We thus have high consistency, but not a good semantic representation. The word *verb* is an example that does work: all nearest neighbours from the background space are linguistic terms. The two *verbs* vectors are also closest to each other (similarity 0.625).

8 Conclusion

Our results show that it is possible to learn consistent embeddings from small data in the context of a low-resource domain, as such data provides consistent contexts to learn from. Applying an additive model that sums general-domain vectors from a pre-trained background space resulted in similar vectors for the same terms across different contexts from the same domain. The Nonce2Vec model also results in consistent embeddings that are closer to vectors of the same term trained on different context sentences than to vectors of other terms. The summed vectors from the additive model applied to our philosophical small data are highly discriminative, distinguishing the target terms from background terms almost perfectly.

Our results show the benefits of using consistency as an intrinsic evaluation metric for distributional semantic models, particularly for low-resource situations in which no gold standard similarity scores are available. While the metric may appear simple, it proved useful both for evaluating the homogeneity of a dataset and for evaluating the stability of vector spaces generated by a given model. Consistency turns out to depend on various combinations of factors, including the nature of the data itself, the model used to train the semantic space, and the frequency of the learned terms, both in the background space and in the in-domain data of interest.

For the specific purpose of modeling philosophical terminology, consistency helps us assess the quality of embeddings for philosophical terms, which may differ in meaning across a book or an author's work, and for which no gold standard evaluation sets are available. These embeddings can then be used to aid in the examination of large volumes of philosophical text (Ginammi et al., in press). Beyond our use case, the consistency metric is quite broadly applicable — a relevant background semantic space is necessary, but this can be constructed from out-of-domain data.

Like any metric, the consistency metric does not answer all of our questions about the quality of our embeddings. Although the additive model is more consistent than the others, both its dependence on data size and the not-always-great qualitative results show that exploring other models is worthwhile for small data. Further research is required to determine whether the representations produced by the additive model are useful for downstream tasks. Using the knowledge of domain experts in a structured evaluation task would be a good, though resource-intensive, next step. Our metric helps quantify the reliability of a model before investing more resources into evaluation.

Our observation that the consistency metric depends on a variety of other factors shows that consistency is a non-trivial aspect of the evaluation of distributional semantic models that should not be overlooked. In future work, we will apply the consistency metric to evaluate other models, and datasets from other domains.

Acknowledgments

We are grateful to Yvette Oortwijn and Arianna Betti for their input as Quine domain experts. We also thank them as well as Lisa Dondorp, Thijs Ossenkoppele and Maud van Lier for their work on the Quine in Context corpus. We thank Pia Sommerauer for help with the SVD setup, and the UvA e-Ideas group for their fruitful discussion of a draft of this paper. We also thank the anonymous reviewers for their time and valuable comments. This research was supported by VICI grant *e-Ideas* (277-20-007) awarded to Arianna Betti and VENI grant *Reading between the lines* 275-89-029 awarded to Antske Fokkens, both financed by the Dutch Research Council (NWO).

References

Fatemeh Torabi Asr, Jon Willits, and Michael Jones. 2016. Comparing predictive and co-occurrence based models of lexical semantics trained on child-directed speech. In *CogSci*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Arianna Betti and Hein van den Berg. 2016. Towards a Computational History of Ideas. In *Proceedings of the Third Conference on Digital Humanities in Luxembourg with a Special Focus on Reading Historical Sources in the Digital Age. CEUR Workshop Proceedings, CEUR-WS.org*, volume 1681, Aachen.

Arianna Betti, Hein van den Berg, Yvette Oortwijn, and Caspar Treijtel. 2019. History of Philosophy in Ones and Zeros. In Mark Curtis and Eugen Fischer, editors, *Methodological Advances in Experimental Philosophy*, Advances in Experimental Philosophy, pages 295–332. Bloomsbury, London.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research (JAIR)*, 49(1-47).

Susan Carey and Elsa Bartlett. 1978. Acquiring a single new word. *Papers and Reports on Child Language Development*, 15:17–29.

Stephen Clark. 2015. Vector space models of lexical meaning. *The Handbook of Contemporary semantic theory*, pages 493–522.

Marc N Coutanche and Sharon L Thompson-Schill. 2014. Fast mapping rapidly integrates information into existing memory networks. *Journal of Experimental Psychology: General*, 143(6):2296.

Katrin Erk. 2012. Vector space models of word meaning and phrase meaning: A survey. *Language and Linguistics Compass*, 6(10):635–653.

Annapaola Ginammi, Jelke Bloem, Rob Koopman, Shenghui Wang, and Arianna Betti. in press. Bolzano, Kant, and the Traditional Theory of Concepts: A Computational Investigation. In *The Dynamics of Science: Computational Frontiers in History and Philosophy of Science*. Pittsburgh University Press.

Anna Gladkova and Aleksandr Drozd. 2016. Intrinsic evaluations of word embeddings: What can we do better? In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 36–42.

Johannes Hellrich and Udo Hahn. 2016. Bad company - neighborhoods in neural embedding spaces considered harmful. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2785–2796.

Aurélie Herbelot and Marco Baroni. 2017. High-risk learning: acquiring new word vectors from tiny data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 304–309.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Alexandre Kabbach, Kristina Gulordava, and Aurélie Herbelot. 2019. Towards incremental learning of word embeddings using context informativeness. In *Proceedings of the 57th Conference of the Association for Computational Linguistics: Student Research Workshop*, pages 162–168, Florence, Italy. Association for Computational Linguistics.

Angeliki Lazaridou, Marco Marelli, and Marco Baroni. 2017. Multimodal word meaning induction from minimal exposure to natural text. *Cognitive science*, 41:677–705.

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.

Thang Luong, Richard Socher, and Christopher Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113.

Matthew McGrath and Devin Frank. 2018. Propositions. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*, spring 2018 edition. Metaphysics Research Lab, Stanford University.

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

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Farhad Nooralahzadeh, Lilja Øvrelid, and Jan Tore Lønning. 2018. Evaluation of domain-specific word embeddings using knowledge resources. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Willard Van Orman Quine. 1934. *A system of logistic*. Harvard University Press.

Willard Van Orman Quine. 1960. *Word & Object*. MIT Press.

Magnus Sahlgren and Alessandro Lenci. 2016. The effects of data size and frequency range on distributional semantic models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 975–980.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 298–307.

Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Guillaume Lample, and Chris Dyer. 2015. Evaluation of word vector representations by subspace alignment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2049–2054.

Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188.

Bin Wang, Angela Wang, Fenxiao Chen, Yuncheng Wang, and C.-C. Jay Kuo. 2019. Evaluating word embedding models: methods and experimental results. *APSIPA Transactions on Signal and Information Processing*, 8:e19.