

High-risk learning: acquiring new word vectors from tiny data

Aurélie Herbelot

Dept. of Translation and Language Sciences
Universitat Pompeu Fabra
aurelie.herbelot@cantab.net

Marco Baroni

Center for Mind/Brain Sciences
University of Trento
marco.baroni@unitn.it

Abstract

Distributional semantics models are known to struggle with small data. It is generally accepted that in order to learn ‘a good vector’ for a word, a model must have sufficient examples of its usage. This contradicts the fact that humans can guess the meaning of a word from a few occurrences only. In this paper, we show that a neural language model such as Word2Vec only necessitates minor modifications to its standard architecture to learn new terms from tiny data, using background knowledge from a previously learnt semantic space. We test our model on word definitions and on a nonce task involving 2-6 sentences’ worth of context, showing a large increase in performance over state-of-the-art models on the definitional task.

1 Introduction

Distributional models (DS: [Turney and Pantel \(2010\)](#); [Clark \(2012\)](#); [Erk \(2012\)](#)), and in particular neural network approaches ([Bengio et al., 2003](#); [Collobert et al., 2011](#); [Huang et al., 2012](#); [Mikolov et al., 2013](#)), do not fare well in the absence of large corpora. That is, for a DS model to learn a word vector, it must have seen that word a sufficient number of times. This is in sharp contrast with the human ability to perform *fast mapping*, i.e. the acquisition of a new concept from a single exposure to information ([Lake et al., 2011](#); [Trueswell et al., 2013](#); [Lake et al., 2016](#)).

There are at least two reasons for wanting to acquire vectors from very small data. First, some words are simply rare in corpora, but potentially crucial to some applications (consider, for instance, the processing of text containing technical

terminology). Second, it seems that fast-mapping should be a prerequisite for any system pretending to cognitive plausibility: an intelligent agent with learning capabilities should be able to make educated guesses about new concepts it encounters.

One way to deal with data sparsity issues when learning word vectors is to use morphological structure as a way to overcome the lack of primary data ([Lazaridou et al., 2013](#); [Luong et al., 2013](#); [Kisselew et al., 2015](#); [Padó et al., 2016](#)). Whilst such work has shown promising result, it is only applicable when there is transparent morphology to fall back on. Another strand of research has been started by [Lazaridou et al. \(2017\)](#), who recently showed that by using simple summation over the (previously learnt) contexts of a nonce word, it is possible to obtain good correlation with human judgements in a similarity task. It is important to note that both these strategies assume that rare words are special cases of the distributional semantics apparatus, and thus require separate approaches to model them.

Having different algorithms for modelling the same phenomenon means however that we need some meta-theory to know when to apply one or the other: it is for instance unclear at which frequency a rare word is not rare anymore. Further, methods like summation are naturally self-limiting: they create frustratingly strong baselines but are too simplistic to be extended and improved in any meaningful way. In this paper, our underlying assumption is thus that it would be desirable to build a single, all-purpose architecture to learn word representations from *any* amount of data. The work we present views fast-mapping as a component of an incremental architecture: the rare word case is simply the first part of the concept learning process, *regardless of how many times it will eventually be encountered*.

With the aim of producing such an incremen-

tal system, we demonstrate that the general architecture of neural language models like Word2Vec (Mikolov et al., 2013) is actually suited to modelling words from a few occurrences only, providing minor adjustments are made to the model itself and its parameters. Our main conclusion is that the combination of a heightened learning rate and greedy processing results in very reasonable one-shot learning, but that some safeguards must be in place to mitigate the high risks associated with this strategy.

2 Task description

We want to simulate the process by which a competent speaker encounters a new word in known contexts. That is, we assume an existing vocabulary (i.e. a previously trained semantic space) which can help the speaker ‘guess’ the meaning of the new word. To evaluate this process, we use two datasets, described below.

The definitional nonce dataset We build a novel dataset based on encyclopedic data, simulating the case where the context of the unknown word is supposedly maximally informative.¹ We first record all Wikipedia titles containing one word only (e.g. *Albedo*, *Insulin*). We then extract the first sentence of the Wikipedia page corresponding to each target title (e.g. *Insulin is a peptide hormone produced by beta cells in the pancreas.*), and tokenise that sentence using the Spacy toolkit.² Each occurrence of the target in the sentence is replaced with a slot (___).

From this original dataset, we only retain sentences with enough information (i.e. a length over 10 words), corresponding to targets which are frequent enough in the UkWaC corpus (Baroni et al. (2009), minimum frequency of 200). The frequency threshold allows us to make sure that we have a high-quality gold vector to compare our learnt representation to. We then randomly sample 1000 sentences, manually checking the data to remove instances that are, in fact, not definitional. We split the data into 700 training and 300 test instances.

On this dataset, we simulate first-time exposure to the nonce word by changing the label of the gold standard vector in the background semantic space, and producing a new, randomly initialised vector

for the nonce. So for instance, *insulin* becomes *insulin_gold*, and a new random embedding is added to the input matrix for *insulin*. This setup allows us to easily measure the similarity of the newly learnt vector, obtained from one definition, to the vector produced by exposure to the whole Wikipedia. To measure the relative performance of various setups, we calculate the Reciprocal Rank (RR) of the gold vector in the list of all nearest neighbours to the learnt representation. We average RRs over the number of instances in the dataset, thus obtaining a single MRR figure (Mean Reciprocal Rank).

The Chimera dataset Our second dataset is the ‘Chimera’ dataset of (Lazaridou et al., 2017).³ This dataset was specifically constructed to simulate a nonce situation where a speaker encounters a word for the first time in naturally-occurring (and not necessarily informative) sentences. Each instance in the data is a nonce, associated with 2-6 sentences showing the word in context. The novel concept is created as a ‘chimera’, i.e. a mixture of two existing and somewhat related concepts (e.g., a buffalo crossed with an elephant). The sentences associated with the nonce are utterances containing one of the components of the chimera, randomly extracted from a large corpus.

The dataset was annotated by humans in terms of the similarity of the nonce to other, randomly selected concepts. Fig. 1 gives an example of a data point with 2 sentences of context, with the nonce capitalised (*VALTUOR*, a combination of *cucumber* and *celery*). The sentences are followed by the ‘probes’ of the trial, i.e. the concepts that the nonce must be compared to. Finally, human similarity responses are given for each probe with respect to the nonce. Each chimera was rated by an average of 143 subjects. In our experiments, we simply replace all occurrences of the original nonce with a slot (___) and learn a representation for that slot. For each setting (2, 4 and 6 sentences), we randomly split the 330 instances in the data into 220 for training and 110 for testing.

Following the authors of the dataset, we evaluate by calculating the correlation between system and human judgements. For each trial, we calculate Spearman correlation (ρ) between the similarities given by the system to each nonce-probe pair, and the human responses. The overall result is the average Spearman across all trials.

¹Data available at <http://aurelieherbelot.net/resources/>.

²<https://spacy.io/>

³Available at <http://clic.cimec.unitn.it/Files/PublicData/chimeras.zip>.

Sentences:
 Canned sardines and VALTUOR between two slices of wholemeal bread and thinly spread Flora Original.
 @@ Erm, VALTUOR, low fat dairy products, incidents of heart disease for those who have an olive oil rich diet.

Probes: rhubarb, onion, pear, strawberry, limousine, cushion
 Human responses: 3, 2.86, 1.43, 2.14, 1.29, 1.71

Figure 1: An example chimera (VALTUOR).

3 Baseline models

We test two state-of-the-art systems: a) Word2Vec (W2V) in its Gensim⁴ implementation, allowing for update of a prior semantic space; b) the additive model of Lazaridou et al. (2017), using a background space from W2V.

We note that both models allow for some sort of incrementality. W2V processes input one context at a time (or several, if mini-batches are implemented), performing gradient descent after each new input. The network’s weights in the input, which correspond to the created word vectors, can be inspected at any time.⁵ As for addition, it also affords the ability to stop and restart training at any time: a typical implementation of this behaviour can be found in distributional semantics models based on random indexing (see e.g. QasemiZadeh et al., 2017). This is in contrast with so-called ‘count-based’ models calculated by computing a frequency matrix over a fixed corpus, which is then globally modified through a transformation such as Pointwise Mutual Information.

Word2Vec We consider W2V’s ‘skip-gram’ model, which learns word vectors by predicting the context words of a particular target. The W2V architecture includes several important parameters, which we briefly describe below.

In W2V, predicting a word implies the ability to distinguish it from so-called *negative samples*, i.e. other words which are *not* the observed item. The number of negative samples to be considered can be tuned. What counts as a context for a particular target depends on the *window size* around that target. W2V features random resizing of the window, which has been shown to increase the model’s performance. Further, each sentence passed to the model undergoes *subsampling*, a random process by which some words are dropped out of the input

⁴Available at <https://github.com/RaRe-Technologies/gensim>.

⁵Technically speaking, standard W2V is not fully incremental, as it requires a first pass through the corpus to compute a vocabulary, with associated frequencies. As we show in §5, it however allows for an incremental interpretation, given minor modifications.

as a function of their overall frequency. Finally, the *learning rate* α measures how quickly the system learns at each training iteration. Traditionally, α is set low (0.025 for Gensim) in order not to overshoot the system’s error minimum.

Gensim has an update function which allows us to save a W2V model and continue learning from new data: this lets us simulate prior acquisition of a background vocabulary and new learning from a nonce’s context. As background vocabulary, we use a semantic space trained on a Wikipedia snapshot of 1.6B words with Gensim’s standard parameters (initial learning rate of 0.025, 5 negative samples, a window of ± 5 words, subsampling $1e^{-3}$, 5 epochs). We use the skip-gram model with a minimum word count of 50 and vector dimensionality 400. This results in a space with 259,376 word vectors. We verify the quality of this space by calculating correlation with the similarity ratings in the MEN dataset (Bruni et al., 2014). We obtain $\rho = 0.75$, indicating an excellent fit with human judgements.

Additive model Lazaridou et al. (2017) use a simple additive model, which sums the vectors of the context words of the nonce, taking as context the entire sentence where the target occurs. Their model operates on multimodal vectors, built over both text and images. In the present work, however, we use the semantic space described above, built on Wikipedia text only. We do not normalise vectors before summing, as we found that the system’s performance was better than with normalisation. We also discard function words when summing, using a stopword list. We found that this step affects results very positively.

The results for our state-of-the-art models are shown in the top sections of Tables 1 and 2. W2V is run with the standard Gensim parameters, under the skip-gram model. It is clear from the results that W2V is unable to learn nonces from definitions ($MRR = 0.00007$). The additive model, on the other hand, performs well: an MRR of 0.03686 means that the median rank of the true vector is 861, out of a challenging

259,376 neighbours (the size of the vocabulary). On the Chimeras dataset, W2V still performs well under the sum model – although the difference is not as marked and possibly indicates that this dataset is more difficult (which we would expect, as the sentences are not as informative as in the encyclopedia case).

4 Nonce2Vec

Our system, Nonce2Vec (N2V),⁶ modifies W2V in the following ways.

Initialisation: since addition gives a good approximation of the nonce word, we initialise our vectors to the sum of all known words in the context sentences (see §3). Note that this is not strictly equivalent to the pure sum model, as subsampling takes care of frequent word deletion in this setup (as opposed to a stopword list). In practice, this means that the initialised vectors are of slightly lesser quality than the ones from the sum model.

Parameter choice: we experiment with higher learning rates coupled with larger window sizes. That is, the model should take the risk of a) overshooting a minimum error; b) greedily considering irrelevant contexts in order to increase its chance to learn anything. We mitigate these risks through *selective training* and appropriate *parameter decay* (see below).

Window resizing: we suppress the random window resizing step when learning the nonce. This is because we need as much data as possible and accordingly need a large window around the target. Resizing would make us run the risk of ending up with a small window of a few words only, which would be uninformative.

Subsampling: With the goal of keeping most of our tiny data, we adopt a subsampling rate that only discards extremely frequent words.

Selective training: we only train the nonce. That is, we only update the weights of the network for the target. This ensures that, despite the high selected learning rate, the previously learnt vectors, associated with the other words in the sentence, will not be radically shifted towards the meaning expressed in that particular sentence.

Whilst the above modifications are appropriate to deal with the first mention of a word, we must ask in what measure they still are applicable when the term is encountered again (see §1). With a

⁶Code available at <https://github.com/minimalparts/nonce2vec>.

	MRR	Median rank
W2V	0.00007	111012
Sum	0.03686	861
N2V	0.04907	623

Table 1: Results on definitional dataset

	L2 ρ	L4 ρ	L6 ρ
W2V	0.1459	0.2457	0.2498
Sum	0.3376	0.3624	0.4080
N2V	0.3320	0.3668	0.3890

Table 2: Results on chimera dataset

view to cater for incrementality, we introduce a notion of **parameter decay** in the system. We hypothesise that the initial high-risk strategy, combining high learning rate and greedy processing of the data, should only be used in the very first training steps. Indeed, this strategy drastically moves the initialised vector to what the system assumes is the right neighbourhood of the semantic space. Once this positioning has taken place, the system should refine its guess rather than wildly moving in the space. We thus suggest that the learning rate itself, but also the subsampling value and window size should be returned to more conventional standards as soon as it is desirable. To achieve this, we apply some exponential decay to the learning rate of the nonce, proportional to the number of times the term has been seen: every time t that we train a pair containing the target word, we set α to $\alpha_0 e^{-\lambda t}$, where α_0 is our initial learning rate. We also decrease the window size and increase subsampling rate on a per-sentence basis (see §5).

5 Experiments

We first tune N2V’s initial parameters on the training part of the definitional dataset. We experiment with a range of values for the learning rate ([0.5, 0.8, 1, 2, 5, 10, 20]), window size ([5, 10, 15, 20]), the number of negative samples ([3, 5, 10]), the number of epochs ([1, 5]) and the subsampling rate ([500, 1000, 10000]). Here, given the size of the data, the minimum frequency for a word to be considered is 1. The best performance is obtained for a window of 15 words, 3 negative samples, a learning rate of 1, a subsampling rate of 10000, an exponential decay where $\lambda = \frac{1}{70}$, and one single epoch (that is, the system truly implements fast-mapping). When applied to

the test set, N2V shows a dramatic improvement in performance over the simple sum model, reaching $MMR = 0.04907$ (median rank 623).

On the training set of the Chimeras, we further tune the per-sentence decrease in window size and increase in subsampling. For the window size, we experiment with a reduction of [1...6] words on either side of the target, not going under a window of ± 3 words. Further, we adjust each word's subsampling rate by a factor in the range [1.1, 1.2...1.9, 2.0]. Our results confirm that indeed, an appropriate change in those parameters is required: keeping them constant results in decreasing performance as more sentences are introduced. On the training set, we obtain our best performance (averaged over the 2-, 4- and 6-sentences datasets) for a per-sentence window size decrease of 5 words on either side of the target, and adjusting subsampling by a factor of 1.9. Table 2 shows results on the three corresponding test sets using those parameters. Unfortunately, on this dataset, N2V does not improve on addition.


The difference in performance between the definitional and the Chimeras datasets may be explained in two ways. First, the chimera sentences were randomly selected and thus, are not necessarily hugely informative about the nature of the nonce. Second, the most informative sentences are not necessarily at the beginning of the fragment, so the system heightens its learning rate on the wrong data: the risk does not pay off. This suggests that a truly intelligent system should adjust its parameters in a non-monotonic way, to take into account the quality of the information it is processing. This point seems to be an important general requirement for any architecture that claims incrementality: our results indicate very strongly that a notion of *informativeness* must play a role in the learning decisions of the system. This conclusion is in line with work in other domains, e.g. interactive word learning using dialogue, where performance is linked to the ability of the system to measure its own confidence in particular pieces of knowledge and ask questions with a high information gain (Yu et al., 2016). It also meets with general considerations on language acquisition, which accounts for the ability of young children to learn from limited 'primary linguistic data' by restricting explanatory models to those that provide such efficiency (Clark and Lappin, 2010).

6 Conclusion

We have proposed Nonce2Vec, a Word2Vec-inspired architecture to learn new words from tiny data. It requires a high-risk strategy combining heightened learning rate and greedy processing of the context. The particularly good performance of the system on definitions makes us confident that it is possible to build a unique, unified algorithm for learning word meaning from any amount of data. However, the less impressive performance on naturally-occurring sentences indicates that an ideal system should modulate its learning as a function of the informativeness of a context sentence, that is, take risks 'at the right time'.

As pointed out in the introduction, Nonce2Vec is designed with a view to be an essential component of an incremental concept learning architecture. In order to validate our system as a suitable, generic solution for word learning, we will have to test it on various data sizes, from the type of low- to middle-frequency terms found in e.g. the Rare Words dataset (Luong et al., 2013), to highly frequent words. We would like to systematically evaluate, in particular, how fast the system can gain an understanding of a concept which is fully equivalent to a vector built from big data. We believe that both quality and speed of learning will be strongly influenced by the ability of the algorithm to detect what we called *informative* sentences. Our future work will thus investigate how to capture and measure informativeness.

Acknowledgments

We are grateful to Katrin Erk for inspiring conversations about tiny data and fast-mapping, and to Raffaella Bernardi and Sandro Pezzelle for comments on an early draft of this paper. We also thank the anonymous reviewers for their time and valuable comments. We acknowledge ERC 2011 Starting Independent Research Grant No 283554 (COMPOSES). This project has also received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 751250. 

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.
- Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3:1137–1155.
- Elia Bruni, Nam Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49(1):1–47.
- Alexander Clark and Shalom Lappin. 2010. Computational learning theory and language acquisition. In Ruth M Kempson, Tim Fernando, and Nicholas Asher, editors, *Philosophy of linguistics*, pages 445–475. Elsevier.
- Stephen Clark. 2012. Vector space models of lexical meaning. In Shalom Lappin and Chris Fox, editors, *Handbook of Contemporary Semantics – second edition*. Wiley-Blackwell.
- Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.
- Katrin Erk. 2012. Vector space models of word meaning and phrase meaning: a survey. *Language and Linguistics Compass*, 6:635–653.
- Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL2012)*, pages 873–882.
- Max Kisselew, Sebastian Padó, Alexis Palmer, and Jan Šnajder. 2015. Obtaining a Better Understanding of Distributional Models of German Derivational Morphology. In *Proceedings of the 11th International Conference on Computational Semantics (IWCS2015)*, pages 58–63, London, UK.
- Brenden M Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua B Tenenbaum. 2011. One-shot learning of simple visual concepts. In *Proceedings of the 33rd Annual Meeting of the Cognitive Science Society (CogSci2012)*, Boston, MA.
- Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. 2016. **Building machines that learn and think like people**. *arxiv*, abs/1604.00289.
- Angeliki Lazaridou, Marco Marelli, and Marco Baroni. 2017. Multimodal word meaning induction from minimal exposure to natural text. *Cognitive Science*, 41(S4):677–705.
- Angeliki Lazaridou, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2013. Compositionally Derived Representations of Morphologically Complex Words in Distributional Semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL2013)*, pages 1517–1526, Sofia, Bulgaria.
- Thang Luong, Richard Socher, and Christopher D. Manning. 2013. Better Word Representations with Recursive Neural Networks for Morphology. In *Proceedings of the 17th Conference on Computational Natural Language Learning (CoNLL2013)*, pages 104–113, Sofia, Bulgaria.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.
- Sebastian Padó, Aurélie Herbelot, Max Kisselew, and Jan Šnajder. 2016. Predictability of distributional semantics in derivational word formation. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING2016)*, Osaka, Japan.
- Behrang QasemiZadeh, Laura Kallmeyer, and Aurélie Herbelot. 2017. Non-Negative Randomized Word Embeddings. In *Proceedings of Traitement automatique des langues naturelles (TALN2017)*, Orléans, France.
- John C Trueswell, Tamara Nicol Medina, Alon Hafri, and Lila R Gleitman. 2013. Propose but verify: Fast mapping meets cross-situational word learning. *Cognitive psychology*, 66(1):126–156.
- Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188.
- Yanchao Yu, Arash Eshghi, and Oliver Lemon. 2016. Training an adaptive dialogue policy for interactive learning of visually grounded word meanings. In *Proceedings of the 17th Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL2016)*, pages 339–349, Los Angeles, CA.