

Context-Aware Prediction of Derivational Word-forms

Ekaterina Vylomova¹, Ryan Cotterell², Timothy Baldwin¹ and Trevor Cohn¹

¹Department of Computing and Information Systems, The University of Melbourne

²Center for Language and Speech Processing, Johns Hopkins University

{evylomova, ryan.cotterell}@gmail.com

{tbaldwin, tcohn}@unimelb.edu.au

Abstract

Derivational morphology is a fundamental and complex characteristic of language. In this paper we propose the new task of predicting the derivational form of a given base-form lemma that is appropriate for a given context. We present an encoder-decoder style neural network to produce a derived form character-by-character, based on its corresponding character-level representation of the base form and the context. We demonstrate that our model is able to generate valid context-sensitive derivations from known base forms, but is less accurate under a lexicon agnostic setting.

1 Introduction

Understanding how new words are formed is a fundamental task in linguistics and language modelling, with significant implications for tasks with a generation component, such as abstractive summarisation and machine translation. In this paper we focus on modelling derivational morphology, to learn, e.g., that the appropriate derivational form of the verb *succeed* is *succession* given the context *As third in the line of ____...*, but is *success* in *The play was a great ____*.

English is broadly considered to be a morphologically impoverished language, and there are certainly many regularities in morphological patterns, e.g., the common usage of *-able* to transform a verb into an adjective, or *-ly* to form an adverb from an adjective. However there is considerable subtlety in English derivational morphology, in the form of: (a) idiosyncratic derivations; e.g. *picturesque* vs. *beautiful* vs. *splendid* as adjectival forms of the nouns *picture*, *beauty* and *splendour*, respectively; (b) derivational generation in context, which requires the automatic determination of the part-

of-speech (POS) of the stem and the likely POS of the word in context, and POS-specific derivational rules; and (c) multiple derivational forms often exist for a given stem, and these must be selected between based on the context (e.g. *success* and *succession* as nominal forms of *success*, as seen above). As such, there are many aspects that affect the choice of derivational transformation, including morphotactics, phonology, semantics or even etymological characteristics. Earlier works (Thorndike, 1941) analysed ambiguity of derivational suffixes themselves when the same suffix might present different semantics depending on the base form it is attached to (cf. *beautiful* vs. *cupful*). Furthermore, as Richardson (1977) previously noted, even words with quite similar semantics and orthography such as *horror* and *terror* might have non-overlapping patterns: although we observe regularity in some common forms, for example, *horrify* and *terrify*, and *horrible* and *terrible*, nothing tells us why we observe *terrorize* and no instances of *horrorize*, or *horrid*, but not *terrific*.

In this paper, we propose the new task of predicting a derived form from its context and a base form. Our motivation in this research is primarily linguistic, i.e. we measure the degree to which it is possible to predict particular derivation forms from context. A similar task has been proposed in the context of studying how children master derivations (Singson et al., 2000). In their work, children were asked to complete a sentence by choosing one of four possible derivations. Each derivation corresponded either to a noun, verb, adjective, or adverbial form. Singson et al. (2000) showed that childrens' ability to recognize the correct form correlates with their reading ability. This observation confirms an earlier idea that orthographical regularities provide a clearer clues to morphological transformations comparing to phonological rules (Templeton, 1980; Moskowitz, 1973), especially in lan-

guages such as English where grapheme-phoneme correspondences are opaque. For this reason we consider orthographic rather than phonological representations.

In our approach, we test how well models incorporating distributional semantics can capture derivational transformations. Deep learning models capable of learning real-valued word embeddings have been shown to perform well on a range of tasks, from language modelling (Mikolov et al., 2013a) to parsing (Dyer et al., 2015) and machine translation (Bahdanau et al., 2015). Recently, these models have also been successfully applied to morphological reinflexion tasks (Kann and Schütze, 2016; Cotterell et al., 2016a).

2 Derivational Morphology

Two important goals of morphology, the linguistic study of the internal structure of words, are to describe the relation between different words in the lexicon and to decompose them into *morphemes*, the smallest linguistic unit bearing meaning. Morphology can be divided into two types: *inflectional* and *derivational*. Inflectional morphology is the set of processes through which the word form outwardly displays syntactic information, e.g., verb tense. It follows that an inflectional affix typically neither changes the part-of-speech (POS) nor the semantics of the word. For example, the English verb *to run* takes various forms: *run*, *runs* and *ran*, all of which convey the concept “moving by foot quickly”, but appear in complementary syntactic contexts.

Derivation, on the other hand, deals with the formation of new words that have semantic shifts in meaning (often including POS) and is tightly intertwined with lexical semantics (Light, 1996). Consider the example of the English noun *discontentedness*, which is derived from the adjective *discontented*. It is true that both words share a close semantic relationship, but the transformation is clearly more than a simple inflectional marking of syntax. Indeed, we can go one step further and define a chain of words *content* \mapsto *contented* \mapsto *discontented* \mapsto *discontentedness*.

In this work, we deal with the formation of deverbal nouns, i.e., nouns that are formed from verbs. Common examples of this in English include agentives, e.g., *explain* \mapsto *explainer*, gerunds, e.g., *explain* \mapsto *explaining* as well as other nominalizations, e.g., *explain* \mapsto *explanation*. These nominal-

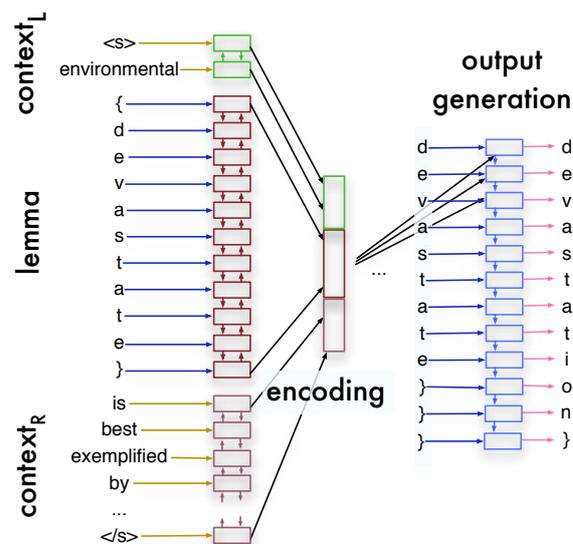


Figure 1: The encoder-decoder model, showing the stem *devastate* in context producing the form *devastation*. Coloured arrows indicate shared parameters.

izations all have drastically different meanings—a key focus of our study is the prediction of which form is most appropriate depending on the context. We expect that our model is general and could be deployed on other related lexical problems.

3 Related Work

Although in the last few years many neural morphological models have been proposed, most of them have focused on inflectional morphology; see Cotterell et al. (2016a) for an overview of the state of the art. Focusing on derivational processes, there are three main directions of research. The first deals with evaluation of word embeddings either using a word analogy task (Gladkova et al., 2016) or binary relation type classification (Vylomova et al., 2016). And it has been shown that, unlike inflectional morphology, most derivational relations cannot be as easily captured using distributional methods. Researchers working on the second type of task attempt to predict derived forms using the embedding of its corresponding base form and a vector encoding a “derivational” shift. Guevara (2011) notes that derivational affixes can be modelled as a geometrical function over the vectors of the base forms. On the other hand, Lazaridou et al. (2013) and Cotterell and Schütze (2017) represent derivational affixes as vectors and investigate various functions to combine them with base forms. Kisselew et al. (2015) and Padó et al. (2016) extend this line of research to model derivational

morphology in German. Their study demonstrates that various factors such as part of speech, semantic regularity and argument structure (Grimshaw, 1990) influence predictability of a derived word. The third area of research focuses on the analysis of derivationally complex forms, which differs from study in that we focus on generation. The goal of this line of work is to produce a canonicalized segmentation of an input word into its constituent morphs, e.g., *unhappiness* \rightarrow *un+happy+ness* (Cotterell et al., 2015; Cotterell et al., 2016b). Note that the orthographic change $y \rightarrow i$ has been reversed.

4 Dataset

As the starting point for the construction of our dataset, we used the CELEX English dataset (Baayen et al., 1993). We extracted verb–noun lemma pairs from CELEX, covering 24 different nominalisational suffixes and 1,456 base lemmas. Suffixes only occurring in 5 or fewer lemma pairs mainly corresponded to loan words and consequently were filtered out. We augmented this dataset with verb–verb pairs, one for each verb present in the verb–noun pairs, to capture the case of a verbal form being appropriate for the given context.¹ For each noun and verb lemma, we generated all their inflections, and searched for sentential contexts of each inflected token in a pre-tokenised dump of English Wikipedia.² To dampen the effect of high-frequency words, we applied a heuristic log function threshold which is basically a weighted logarithm of the number of the contexts. The final dataset contains 3,079 unique lemma pairs represented in 107,041 contextual instances.³

5 Experiments

In this paper we model derivational morphology as a prediction task, formulated as follows. We take sentences containing a derivational form of a given lemma, then obscure the derivational form by replacing it with its base form lemma. The system must then predict the original (derivational) form, which may make use of the sentential context. System predictions are judged correct if they exactly match the original derived form.

¹We experimented without verb–verb pairs and didn’t observe much difference in the results.

²Based on a 2008/03/12 dump. Sentences shorter than 3 words or longer than 50 words were removed from the dataset.

³The code and the dataset are available at <https://github.com/ivri/dmorph>

5.1 Baseline

As a baseline we considered a trigram model with modified Kneser-Ney smoothing, trained on the training partition. Each sentence in the testing data was augmented with a set of confabulated sentences, where we replaced a target word with other its derivations or a base form. Unlike the general task, where we generate word forms as character sequences, here we use a set of known inflected forms for each lemma. We then use the language model to score the collections of test sentences, and selected the variant with the highest language model score, and evaluate accuracy of selecting the original word form.

5.2 Encoder-Decoder Model

We propose an encoder-decoder model. The encoder combines the left and the right contexts as well as a character-level base form representation:

$$t = \max(0, H \cdot [h_{\text{left}}^{\rightarrow}; h_{\text{left}}^{\leftarrow}; h_{\text{right}}^{\rightarrow}; h_{\text{right}}^{\leftarrow}; h_{\text{base}}^{\rightarrow}; h_{\text{base}}^{\leftarrow}] + b_h), \quad (1)$$

where $h_{\text{left}}^{\rightarrow}$, $h_{\text{left}}^{\leftarrow}$, $h_{\text{right}}^{\rightarrow}$, $h_{\text{right}}^{\leftarrow}$, $h_{\text{base}}^{\rightarrow}$, $h_{\text{base}}^{\leftarrow}$ correspond to the last hidden states of an LSTM (Hochreiter and Schmidhuber, 1997) over left and right contexts and the character-level representation of the base form (in each case, applied forwards and backwards), respectively; $H \in \mathbb{R}^{[h \times l \times 1.5, h \times l \times 6]}$ is a weight matrix, and $b_h \in \mathbb{R}^{[h \times l \times 1.5]}$ is a bias term. $[\cdot]$ denotes a vector concatenation operation, h is the hidden state dimensionality, and l is the number of layers.

Next we add an extra affine transformation, $o = T \cdot t + b_o$, where $T \in \mathbb{R}^{[h \times l \times 1.5, h \times l]}$ and $b_o \in \mathbb{R}^{[h \times l]}$, then o is then fed into the decoder:

$$g(c_{j+1} | c_j, o, l_{j+1}) = \text{softmax}(R \cdot c_j + \max(B \cdot o, S \cdot l_{j+1}) + b_d), \quad (2)$$

where c_j is an embedding of the j -th character of the derivation, l_{j+1} is an embedding of the corresponding base character, B, S, R are weight matrices, and b_d is a bias term.

We now elaborate on the design choices behind the model architecture, which have been tailored to our task. We supply the model with the l_{j+1} character to enable a copying mechanism, otherwise the model might produce a word prefix which is different from its stem. In most cases the derived form is longer than its stem, and accordingly, when we reach the end of the base form, we continue to

	Shared	Split
baseline	0.63	-
biLSTM+BS	0.58	0.36
biLSTM+CTX	0.80	0.45
biLSTM+CTX+BS	0.83	0.52
biLSTM+CTX+BS+POS	0.89	0.63
LSTM+CTX+BS+POS	0.90	0.66

Table 1: Accuracy for predicted lemmas (bases and derivations) on shared and split lexicons

input a end-of-word symbol. We provide the model with the context vector \mathbf{o} at each decoding step. It has been previously shown (Hoang et al., 2016) that this yields better results than other means of incorporation.⁴ Finally, we use max pooling to enable the model to switch between copying of a stem or producing a new character.

5.3 Settings

We used a 3-layer bidirectional LSTM network, with hidden dimensionality h for both context and base-form stem states of 100, and character embedding c_j of 100.⁵ We used pre-trained 300-dimensional Google News word embeddings (Mikolov et al., 2013a; Mikolov et al., 2013b). During the training of the model, we keep the word embeddings fixed, for greater applicability to unseen test instances. All tokens that didn’t appear in this set were replaced with UNK sentinel tokens. The network was trained using SGD with momentum until convergence.

5.4 Results

With the encoder-decoder model, we experimented with the encoder-decoder, “biLSTM+CTX+BS”, as described in §5.2, as well as several variations, namely: excluding context information (“biLSTM+BS”), and excluding the bidirectional stem (“biLSTM+CTX”). We also investigated how much improvement we can get from knowing the POS tag of the derived form, by presenting it explicitly to the model as extra conditioning context (“biLSTM+CTX+BS+POS”). The main motivation for this relates to gerunds, where without the POS, the model often overgenerates nominalisations. We then tried a single-directional context representation, by using only the last hidden states, i.e., $\mathbf{h}_{\text{left}}^{\rightarrow}$ and $\mathbf{h}_{\text{right}}^{\leftarrow}$, corresponding to the words to

⁴We tried to feed the context information at the initial step only, and this led to worse prediction in terms of context-aware suffixes.

⁵We also experimented with 15 dimensions, but found this model to perform worse.

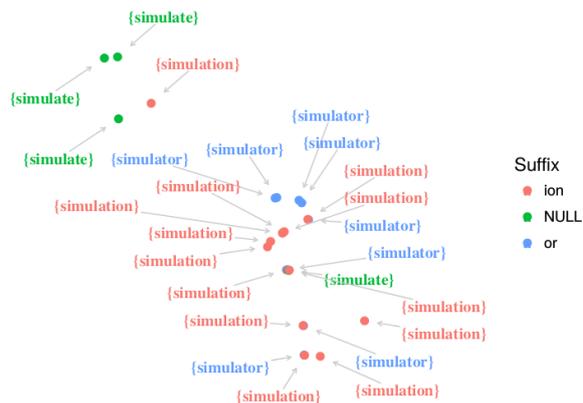


Figure 2: An example of t-SNE projection (Maaten and Hinton, 2008) of context representations for “simulate” base form.

the immediate left and right of the wordform to be predicted (“LSTM+CTX+BS+POS”).

We ran two experiments: first, a shared lexicon experiment, where every stem in the test data was present in the training data; and second, using a split lexicon, where every stem in the test data was *unseen* in the training data. The results are presented in Table 1, and show that: context has a strong impact on results, particularly in the shared lexicon case; there is strong complementarity between the context and character representations, particularly in the split lexicon case; and POS information is particularly helpful in the split lexicon case. Note that most of the models significantly outperform our baseline under shared lexicon setting. The baseline model doesn’t allow split lexicon setting, so we cannot do any comparison at this point.

5.5 Error Analysis

We carried out error analysis over the produced forms of the LSTM+CTX+BS+POS model. First, the model sometimes struggles to differentiate between nominal suffixes: in some cases it puts an agentive suffix (*-er* or *-or*) in contexts where a non-agentive nominalisation (e.g. *-ation* or *-ment*) is appropriate. As an illustration of this, Figure 2 is a t-SNE projection of the context representations for *simulate* vs. *simulator* vs. *simulation*, showing that the different nominal forms have strong overlap. Secondly, although the model learns whether to copy or produce a new symbol well, some forms are spelled incorrectly. Examples of this are *studint*, *studion* or even *studyant* rather than *student* as the agentive nominalisation of *study*. Here, the issue is opaqueness in the etymology, with *student* be-

affix	Re	affix	Re	affix	Re	affix	Re
-age	.93	-al	.95	-ance	.75	-ant	.65
-ation	.93	-ator	.77	-ee	.52	-ence	.82
-ent	.65	-er	.87	-ery	.84	-ion	.93
-ist	.80	-ition	.89	-ment	.90	-or	.64
-th	.95	-ure	.77	-y	.83	NULL	.98

Table 2: Recall for various suffix types. Here *NULL* corresponds to verb-verb cases.

ing borrowed from the Old French *estudiant*. For transformations which are native to English, for example, *-ate* \mapsto *-ation*, the model is much more accurate. Table 2 shows recall values achieved for various suffix types. We do not present precision since it could not be reliably estimated without extensive manual analysis.

In the split lexicon setting, the model sometimes misses double consonants at the end of words, producing *wraper* and *winer* and is biased towards generating mostly productive suffixes. An example of the last case might be *stoption* in place of *stoppage*. We also studied how much the training size affects the model’s accuracy by reducing the data from 1,000 to 60,000 instances (maintaining a balance over lemmas). Interestingly, we didn’t observe a significant reduction in accuracy. Finally, note that under split lexicon setting the model is agnostic of existing derivations, sometimes over-generating possible forms. A nice illustration of that is *trailation*, *trailment* and *trailer* all being produced in the contexts of *trailer*. In other cases, the model might miss some of the derivations, for instance, predicting only *government* in the contexts of *governance* and *government*. We hypothesize that it is either due to very subtle differences in their contexts or higher productivity of *ment*.

Finally, we experimented with some nonsense stems, overwriting sentential instances of *transcribe* to generate context-sensitive derivational forms. Table 3 presents the nonsense stems, the correct form of *transcribe* for a given context, and the predicted derivational form of the nonsense word. Note that the base form is used correctly (top row) for three of the four nonsense words, and that despite the wide variety of output forms they resemble plausible words in English. By looking at a larger slice of the data, we observed some regularities. For instance, *fapery* was mainly produced in the contexts of *transcript* whereas *fapication* was more related to *transcription*. Table 3 also shows that some of the stems appear to be more productive than others.

transcribe	laptify	fape	crimmler	beteption
transcribe	laptify	fape	crimmler	beteption
transcription	laptification	fapery	crimmler	beteption
transcription	laptification	fapication	crimmler	beteption
transcription	laptification	fapionment	crimmler	beteption
transcription	laptification	fapist	crimmler	beteption
transcript	laptification	fapery	crimmler	beteption
transcript	laptification	fapist	crimmler	beteption

Table 3: An experiment with nonsense base forms used in the contexts of *transcribe*.

6 Conclusion and Future Work

We investigated the novel task of context-sensitive derivation prediction for English, and proposed an encoder-decoder model to generate nominalisations. Our best model achieved an accuracy of 90% on a shared lexicon, and 66% on a split lexicon. This suggests that there is regularity in derivational processes and, indeed, in many cases the context is indicative. As we mentioned earlier, there are still many open questions which we leave for future studies. Further, we plan to scale to other languages and augment our dataset with Wiktionary data, to realise a much larger coverage and variety of derivational forms.

7 Acknowledgments

We would like to thank all reviewers for their valuable comments and suggestions. The second author was supported by a DAAD Long-Term Research Grant and an NDSEG fellowship. This research was supported in part by the Australian Research Council.

References

- Harald R Baayen, Richard Piepenbrock, and H van Rijn. 1993. The CELEX lexical data base on CD-ROM.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*, volume abs/1409.0473.
- Ryan Cotterell and Hinrich Schütze. 2017. Joint semantic synthesis and morphological analysis of the derived word. *CoRR*, abs/1701.00946.
- Ryan Cotterell, Thomas Müller, Alexander Fraser, and Hinrich Schütze. 2015. Labeled morphological segmentation with semi-markov models. In *Proceedings of the 19th Conference on Computational Natural Language Learning (CoNLL 2015)*, pages 164–174.

- Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016a. The SIGMORPHON 2016 shared task morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 10–22.
- Ryan Cotterell, Tim Vieira, and Hinrich Schütze. 2016b. A joint model of orthography and morphological segmentation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2016)*, pages 664–669.
- Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)*. abs/1505.08075.
- Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuo. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2016)*, pages 8–15.
- Jane Grimshaw. 1990. *Argument structure*. The MIT Press, Cambridge, MA, US.
- Emiliano Guevara. 2011. Computing semantic compositionality in distributional semantics. In *Proceedings of the 9th International Conference on Computational Semantics*, pages 135–144. Association for Computational Linguistics.
- Cong Duy Vu Hoang, Trevor Cohn, and Gholamreza Haffari. 2016. Incorporating side information into recurrent neural network language models. In *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2016)*, pages 1250–1255.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Katharina Kann and Hinrich Schütze. 2016. Single-model encoder-decoder with explicit morphological representation for reinflection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*.
- 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 (IWCS 2015)*, pages 58–63.
- 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 (ACL 2013)*, pages 1517–1526.
- Marc Light. 1996. Morphological cues for lexical semantics. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL 1996)*, pages 25–31.
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of the Workshop at the International Conference on Learning Representations, 2013*.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of Neural Information Processing Systems Conference (NIPS 2013)*, pages 3111–3119.
- Arlene Moskowitz. 1973. On the status of vowel shift in English. In Timothy E. Moore, editor, *Cognitive Development and the Acquisition of Language*. Academic Press.
- 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 (COLING 2016)*, pages 1285–1297.
- John TE Richardson. 1977. Lexical derivation. *Journal of Psycholinguistic Research*, 6(4):319–336.
- Maria Singson, Diana Mahony, and Virginia Mann. 2000. The relation between reading ability and morphological skills: Evidence from derivational suffixes. *Reading and writing*, 12(3):219–252.
- Shane Templeton. 1980. Spelling, phonology, and the older student. *Developmental and cognitive aspects of learning to spell: A reflection of word knowledge*, pages 85–96.
- Edward Lee Thorndike. 1941. *The teaching of English suffixes*, volume 847. Teachers College, Columbia University.
- Ekaterina Vylomova, Laura Rimmel, Trevor Cohn, and Timothy Baldwin. 2016. Take and took, gaggle and goose, book and read: evaluating the utility of vector differences for lexical relation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pages 1671–1682.