SimpleScience: Lexical Simplification of Scientific Terminology

Yea-Seul Kim and **Jessica Hullman**
University Washington
Information School
yeaseul1, jhullman@uw.edu

Matthew Burgess
University of Michigan
Computer Science Department
mattburg@umich.edu

Eytan Adar
University of Michigan
School of Information
eadar@umich.edu

Abstract

Lexical simplification of scientific terms represents a unique challenge due to the lack of a standard parallel corpora and fast rate at which vocabulary shift along with research. We introduce SimpleScience, a lexical simplification approach for scientific terminology. We use word embeddings to extract simplification rules from a parallel corpora containing scientific publications and Wikipedia. To evaluate our system we construct SimpleSciGold, a novel gold standard set for science-related simplifications. We find that our approach outperforms prior context-aware approaches at generating simplifications for scientific terms.

1 Introduction

Lexical simplification, the process of reducing the complexity of words by replacing them with simpler substitutes (e.g., *sodium* in place of *Na*; *insects* in place of *lepidopterans*) can make scientific texts more accessible to general audiences. Human-in-the-loop interfaces present multiple possible simplifications to a reader (on demand) in place of jargon and give the reader familiar access points to understanding jargon (Kim et al., 2015). Unfortunately, simplification techniques are not yet of high enough quality for fully automated scenarios.

Currently lexical simplification pipelines for scientific texts are rare. The vast majority of prior methods assume a domain independent context, and rely on Wikipedia and Simple English Wikipedia, a subset of Wikipedia using simplified grammar and terminology, to learn simplifications (Biran et al., 2011; Paetzold and Specia, 2015), with translation-based approaches using an aligned version (Coster and Kauchak, 2011; Horn et al., 2014; Yatskar et al., 2010). However, learning simplifications from Wikipedia is not well suited to lexical simplification of scientific terms. Though generic or established terms may appear in Wikipedia, novel terms associated with new advances may not be reflected. Wikipedia's editing rules also favor generality over specificity and eliminate redundancy, both of which are problematic in providing a rich training set that matches simple and complex terms. Further, some approaches work by detecting all pairs of words in a corpus and filtering to isolate synonym or hypernym-relationship pairs using WordNet (Biran et al., 2011). Like Wikipedia, WordNet is a general purpose semantic database (Miller, 1995), and does not cover all branches of science nor integrate new terminology quickly.

Word embeddings do not require the use of prebuilt ontologies to identify associated terms like simplifications. Recent work indicates that they may improve results for simplification *selection*: determining which simplifications for a given complex word can be used without altering the meaning of the text (Paetzold and Specia, 2015). Embeddings have also been explored to extract hypernym relations from general corpora (Rei and Briscoe, 2014). However, word embeddings have not been used for *generating* lexical simplifications. We provide a novel demonstration of how using embeddings on a scientific corpus is better suited to learning scientific term simplifications than prior approaches that use WordNet as a filter and Wikipedia as a corpus.

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1066–1071,
Austin, Texas, November 1-5, 2016.

INPUT: Finally we show that the transient immune activation that renders mosquitoes resistant to the human malaria parasite has little to no effect on mosquito fitness as a measure of survival or fecundity under laboratory conditions. **CANDIDATE RULES**: {fecundity→fertility} {fecundity→productivity} **OUTPUT**: Finally we show that the transient immune activation that renders mosquitoes resistant to the human malaria parasite has little to no effect on mosquito fitness as a measure of survival or (**fertility**; **productivity**) under laboratory conditions.

Table 1: Input sentence, candidate rules and output sentence. (Further examples provided as supplementary material.)

We introduce SimpleScience, a novel lexical simplification pipeline for scientific terms, which we apply to a scientific corpus of nearly 500k publications in Public Library of Science (PLOS) and PubMed Central (PMC) paired with a general corpus from Wikipedia. We validate our approach using SimpleSciGold, a gold standard set that we create using crowdsourcing that contains 293 sentences containing scientific terms with an average of 21 simplifications per term. We show how the SimpleScience pipeline achieves better performance (F-measure: 0.285) than the prior approach to simplification generation applied to our corpus (F-measure: 0.136). We further find that the simplification selection techniques used in prior work to determine which simplifications are a good fit for a sentence do not improve performance when our generation pipeline is used. [1]

2 Parallel corpora: Scientific and General

We assembled a *scientific corpus* of papers from the entire catalog of PLOS articles and the National Library of Medicine's Pubmed Central (PMC) archive (359,324 fulltext articles). The PLOS corpus of 125,378 articles includes articles from PLOS One and each speciality PLOS journal (e.g., Pathogens, Computational Biology). Our *general corpus* includes all 4,776,093 articles from the Feb. 2015 English Wikipedia snapshot. We chose Wikipedia as it covers many scientific concepts and usually contains descriptions of those concepts using simpler language than the research literature. We obtained all datasets from DeepDive (Ré and Zhang, 2015).

[1]Data and source code are available at: https://github.com/yeaseulkim/SimpleScience

3 SimpleScience Design

3.1 Generating Simplifications

Our goal is to learn simplification rules in the form *complex word→simple word*. One approach identifies all pairwise permutations of 'content' terms and then applies semantic (i.e., WordNet) and simplicity filters to eliminate pairs that are not simplifications(Biran et al., 2011). We adopt a similar pipeline but leverage distance metrics on word embeddings and a simpler frequency filter in place of WordNet. Embeddings identify words that share context in an unsupervised, scalable way and are more efficient than constructing co-occurrence matrices (Biran et al., 2011). As our experiments demonstrate, our approach improves performance on a scientific test set over prior work.

3.1.1 Step 1: Generating Word Embeddings

We used the Word2Vec system (Mikolov et al., 2013) to learn word vectors for each content word in the union of vocabulary of the scientific and general corpus. While other approaches exist (Pennington et al., 2014; Levy and Goldberg, 2014), Word2Vec has been shown to produce both fast and accurate results (Mikolov et al., 2013). We set the embedding *dimension* to 300, the *context-window* to 10, and use the skip-gram architecture with negative-sampling,which is known to produce quality results for rare entities (Mikolov et al., 2013).

3.1.2 Step 2: Filtering Pairs

Given the set of all pairwise permutations of words, we retain a simplification rule of two words w_1, w_2 if the cosine similarity $cos(w_1, w_2)$ between the word vectors is greater than a threshold a. We use grid search, described below, to parameterize a.

We then apply additional filtering rules. To avoid rules comprised of words with the same stem (e.g., *permutable*, *permutation*) we stem all words (using the Porter stemmer in the Python NLTK library (Bird et al., 2009)). The POS of each word is determined by Morphadorner (Burns, 2013) and pairs that differ in POS are omitted (e.g., *permutation* (noun), *change(d)* (verb)); Finally, we omit rules where one word is a prefix of the other and the suffix is one of *s, es, ed, ly, er*, or *ing*.

To retain only rules of the form *complex word* →

simple word we calculate the *corpus complexity*, C (Biran et al., 2011) of each word w as the ratio between the frequency (f) in the scientific versus general corpus: $C_w = f_{w,scientific}/f_{w,general}$. The *lexical complexity*, L, of a word is calculated as the word's character length, and the final complexity of the word as $C_w \times L_w$. We require that the final complexity score of the first word in the rule be greater than the second.

While this simplicity filter has been shown to work well in general corpora (Biran et al., 2011), it is sensitive to very small differences in the frequencies with which both words appear in the corpora. This is problematic given the distribution of terms in our corpora, where many rarer scientific terms may appear in small numbers in both corpora.

We introduce an additional constraint that requires that the second (simple) word in the rule occur in the general corpus at least k times. This helps ensure that we do not label words that are at a similar complexity level as simplifications. We note that this filter aligns with prior work that suggests that features of the hypernym in hypernym-hyponym relations influence performance more than features of the hyponym (Rei and Briscoe, 2014).

Parameterization: We use a grid search analysis to identify which measures of the set including $cos(w_1, w_2)$, $f_{w1,scientific}$, $f_{w2,scientific}$, $f_{w1,general}$, and $f_{w2,general}$ most impact the F-measure when we evaluate our generation approach against our scientific gold standard set (Sec. 4), and to set the specific parameter values. Using this method we identify a=0.4 for cosine similarity and k=3,000 for the frequency of the simple term in the general corpus. Full results are available in supplementary material.

3.2 Applying Simplifications

In prior context-aware simplification systems, the decision of whether to apply a simplification rule in an input sentence is complex, involving several similarity operations on word co-occurrence matrices (Biran et al., 2011) or using embeddings to incorporate co-occurrence context for pairs generated using other means (Paetzold and Specia, 2015). However, the SimpleScience pipline already considers the context of appearance for each word in deriving simplifications via word embeddings learned from a large corpus. We see no additional improvements in F-measure when we apply two variants of context similarity thresholds to decide whether to apply a rule to an input sentence. The first is the cosine similarity between the distributed representation of the simple word and the sum of the distributed representations of all words within a window l surrounding the complex word in the input sentence (Paetzold and Specia, 2015). The second is the cosine similarity of a minimum shared frequency co-occurrence matrix for the words in the pair and the co-occurrence matrix for the input sentence (Biran et al., 2011).

In fully automated applications, the top rule from the ranked candidate rules is used. We find that ranking by the cosine similarity between the word embeddings for the complex and simple word in the rule leads to the best performance at the top slot (full results in supplementary material).

4 Evaluation

4.1 SimpleSciGold Test Set

To evaluate our pipeline, we develop SimpleSciGold, a scientific gold standard set of sentences containing complex scientific terms which is modeled after the general purpose gold standard set created by Horn et al. (2014).

To create SimpleSciGold, we start with scientific terms from two sources: we utilized all 304 complex terms from unigram rules by (Vydiswaran et al., 2014), and added another 34,987 child terms from rules found by mining direct parent-child relations for unigrams in the Medical Subject Headings (MeSH) ontology (United States National Library of Medicine, 2015). We chose complex terms with pre-existing simplifications as it provided a means by which we could informally check the crowd generated simplifications for consistency.

To obtain words in context, we extracted 293 sentences containing unique words in this set from PLOS abstracts from PLOS Biology, Pathology, Genetics, and several other journals. We present 10 MTurk crowdworkers with a task ("HIT") showing one of these sentences with the complex word bolded. Workers are told to read the sentence, consult online materials (we provide direct links to a Wikipedia search, a standard Google search, and

		SimpleSciGold			
Method	**Corpus (Complex, Simple)**	**Number of Simplifications**	**Pot.**	**Prec.**	**F**
Biran et al. 2011	Wikipedia, SEW	17	0.059	0.036	0.044
	PLOS/PMC, Wikip.	588	0.352	0.084	0.136
SimpleScience ($cos \geq .4, f_{w,simple} \geq 3000$)	PLOS/PMC, Wikip.	2,322	0.526	0.196	**0.285**
SimpleScience ($cos \geq .4, f_{w,simple} \geq 0$)	PLOS/PMC, Wikip.	10,910,536	0.720	0.032	0.061

Table 2: Simplification Generation Results. SimpleScience achieves the highest F-measure with a cosine threshold of 0.4 and a frequency of the simple word in the general corpus of 3000.

a Google "define" search on the target term), and add their simplification suggestions. Crowdworkers first passed a multiple choice practice qualification in which they were presented with sentences containing three complex words in need of simplification along with screenshots of Wikipedia and dictionary pages for the terms. The workers were asked to identify which of 5 possible simplifications listed for each complex word would preserve the meaning while simplifying. 108 workers took part in the gold standard set creation task, completing an average of 27 HITs each. The resulting SimpleSciGold standard set consists of an average of 20.7 simplifications for each of the 293 complex words in corresponding sentences.

4.2 Simplification Generation

We compare our word embedding generation process (applied to our corpora) to Biran et al.'s (2011) approach (applied to the Wikipedia and Simple English Wikipedia corpus as well as our scientific corpora). Following the evaluation method used in Paetzold and Specia (2015), we calculate *potential* as the proportion of instances for which at least one of the substitutions generated is present in the gold standard set, *precision* as the proportion of generated instances which are present in the gold standard set, and *F-measure* as their harmonic mean.

Our SimpleScience approach outperforms the original approach by Biran et al. (2011) applied to the Wikipedia and SEW corpus as well as to the scientific corpus (Table 1).

4.3 Applying Simplifications

We find that neither prior selection approaches yield performance improvements over our generation process. We evaluate the performance of ranking candidate rules by cosine similarity (to find the top rule for a fully automated application), and achieve precision of 0.389 at the top slot. In our supplementary materials, we provide additional results for potential, precision and F-measure at varying numbers of slots (up to 5), where we test ranking by cosine similarity of embeddings as well as by the second filter used in our pair generation step: the frequency of the simple word in the simple corpus.

4.4 Antonym Prevalence Analysis

A risk of using Word2Vec in place of WordNet is that the simpler terms generated by our approach may represent terms with opposite meanings (antonyms). While a detailed analysis is beyond the scope of this paper, we compared the likelihood of seeing antonyms in our results using a gold standard set of antonyms for biology, chemistry, and physics terms from WordNik (Wordnik, 2009). Specifically, we created an antonym set consisting of the 304 terms from the biology, chemistry, and physics categories in Wictionary for which at least one antonym is listed in WordNik. We compared antonym pairs with rules that produced by the SimpleScience pipeline (Fig. 1). We observed that 14.5% of the time (44 out of 304 instances), an antonym appeared at the top slot among results. 51.3% of the time (156 out of 304 instances), no antonyms in the list appeared within the top 100 ranked results. These results suggest that further filters are necessary to ensure high enough quality results for fully automated applications of scientific term simplification.

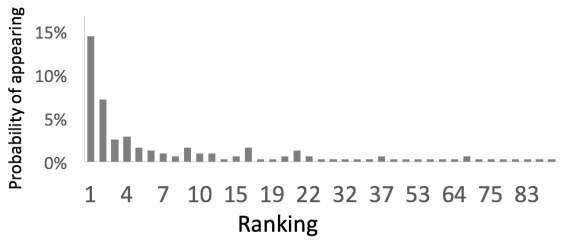

Figure 1: Probability of an antonym in our test set occurring as a suggested simpler term in the top 100 slots in the SimpleScience pipeline.

5 Limitations and Future Work

A risk of using Word2Vec to find related terms, rather than querying a lexical database like WordNet, is that generated rules may include antonyms. Adding techniques to filter antonym rules, such as using co-reference chains (Adel and Schütze, 2014), is important in future work.

We achieve a precision of 0.389 at the top slot on our SimpleSciGold standard set when we apply our generation method and rank candidates by cosine similarity. This level of precision is higher than that achieved by various prior ranking methods used in Lexenstein (Paetzold and Specia, 2015), with the exception of using machine learning techniques like SVM (Paetzold and Specia, 2015). Future work should explore how much the precision of our SimpleScience pipeline can be improved by adopting more sophisticated ranking methods. However, we suspect that even the highest precision obtained on general corpora and gold standard sets in prior work is not sufficient for fully automated simplification. An exciting area for future work is in applying the SimpleScience pipeline in interactive simplification suggestion interfaces for those reading or writing about science (Kim et al., 2015).

6 Conclusion

In this work, we introduce SimpleScience, a lexical simplification approach to address the unique challenges of simplifying scientific terminology, including a lack of parallel corpora, shifting vocabulary, and mismatch with using general purpose databases for filtering. We use word embeddings to extract simplification rules from a novel parallel corpora that contains scientific publications and Wikipedia. Using SimpleSciGold, a gold standard set that we created using crowdsourcing, we show that using embeddings and simple frequency filters on a scientific corpus outperforms prior approaches to simplification generation, and renders the best prior approach to simplification selection unnecessary.

References

[Adel and Schütze2014] Heike Adel and Hinrich Schütze. 2014. Using mined coreference chains as a resource for a semantic task. In *EMNLP*, pages 1447–1452.

[Biran et al.2011] Or Biran, Samuel Brody, and Noémie Elhadad. 2011. Putting it simply: A context-aware approach to lexical simplification. In *ACL '11*. Association for Computational Linguistics.

[Bird et al.2009] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python*. O'Reilly Media, Inc.

[Burns2013] Philip R Burns. 2013. Morphadorner v2: a java library for the morphological adornment of english language texts. *Northwestern University, Evanston, IL*.

[Coster and Kauchak2011] William Coster and David Kauchak. 2011. Learning to simplify sentences using wikipedia. In *Proceedings of the workshop on monolingual text-to-text generation*, pages 1–9. Association for Computational Linguistics.

[Horn et al.2014] Colby Horn, Cathryn Manduca, and David Kauchak. 2014. Learning a lexical simplifier using wikipedia. In *ACL (2)*, pages 458–463.

[Kim et al.2015] Yea-Seul Kim, Jessica Hullman, and Eytan Adar. 2015. Descipher: A text simplification tool for science journalism. In *Computation+Journalism Symposium*.

[Levy and Goldberg2014] Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2177–2185. Curran Associates, Inc.

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

[Miller1995] George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November.

[Paetzold and Specia2015] Gustavo Henrique Paetzold and Lucia Specia. 2015. Lexenstein: A framework for lexical simplification. *ACL-IJCNLP 2015*, 1(1):85.

[Pennington et al.2014] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

[Ré and Zhang2015] Christopher Ré and Ce Zhang. 2015. Deepdive open datasets. `http://deepdive.stanford.edu/opendata`.

[Rei and Briscoe2014] Marek Rei and Ted Briscoe. 2014. Looking for hyponyms in vector space. In *CoNLL*, pages 68–77.

[United States National Library of Medicine2015] United States National Library of Medicine. 2015. Medical subject headings.

[Vydiswaran et al.2014] V.G.Vinod Vydiswaran, Qiaozhu Mei, David A. Hanauer, and Kai Zheng. 2014. Mining consumer health vocabulary from community-generated text. In *AMIA '14*.

[Wordnik2009] Wordnik. 2009. Wordnik online english dictionary. `https://www.wordnik.com/`.

[Yatskar et al.2010] Mark Yatskar, Bo Pang, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2010. For the sake of simplicity: Unsupervised extraction of lexical simplifications from wikipedia. In *NAACL '10*. Association for Computational Linguistics.