# Simplifying Lexical Simplification: Do We Need Simplified Corpora?

**Goran Glavaš**
University of Zagreb
Faculty of Electrical Engineering and Computing
goran.glavas@fer.hr

**Sanja Štajner**
University of Wolverhampton
Research Group in Computational Linguistics
SanjaStajner@wlv.ac.uk

## Abstract

Simplification of lexically complex texts, by replacing complex words with their simpler synonyms, helps non-native speakers, children, and language-impaired people understand text better. Recent lexical simplification methods rely on manually simplified corpora, which are expensive and time-consuming to build. We present an unsupervised approach to lexical simplification that makes use of the most recent word vector representations and requires only regular corpora. Results of both automated and human evaluation show that our simple method is as effective as systems that rely on simplified corpora.

## 1 Introduction

Lexical complexity makes text difficult to understand for various groups of people: non-native speakers (Petersen and Ostendorf, 2007), children (De Belder and Moens, 2010), people with intellectual disabilities (Feng, 2009; Saggion et al., 2015), and language-impaired people such as autistic (Martos et al., 2012), aphasic (Carroll et al., 1998), and dyslexic (Rello, 2012) people. Automatic simplification that replaces complex words with their simpler synonyms is thus needed to make texts more understandable for everyone.

Lexical simplification systems still predominantly use a set of rules for substituting long and infrequent words with their shorter and more frequent synonyms (Devlin and Tait, 1998; De Belder and Moens, 2010). In generating the substitution rules (i.e., finding simple synonyms of a complex word), most systems refer to lexico-semantic resources like WordNet (Fellbaum, 1998). The non-existence of lexicons like WordNet for a vast number of languages diminishes the impact of these simplification methods.

The emergence of the Simple Wikipedia[1] shifted the focus towards the data-driven approaches to lexical simplification, ranging from unsupervised methods leveraging either the metadata (Yatskar et al., 2010) or co-occurrence statistics of the simplified corpora (Biran et al., 2011) to supervised methods learning substitutions from the sentence-aligned corpora (Horn et al., 2014). Using simplified corpora improves the simplification performance, but reduces method applicability to the few languages for which such corpora exist.

The research question motivating this work relates to achieving comparable simplification performance without resorting to simplified corpora or lexicons like WordNet. Observing that "simple" words appear in regular (i.e., "complex", not simplified) text as well, we exploit recent advances in word vector representations (Pennington et al., 2014) to find suitable simplifications for complex words. We evaluate the performance of our resource-light approach (1) automatically, on two existing lexical simplification datasets and (2) manually, via human judgements of grammaticality, simplicity, and meaning preservation. The obtained results support the claim that effective lexical simplification can be achieved without using simplified corpora.

## 2 Related Work

Systems for lexical simplification are still dominantly rule-based, i.e., they rely on a set of substitutions, each consisting of a complex word and its simpler synonym, which are in most cases applied regardless of the context in which the complex word appears. Constructing substitution rules involves identifying synonyms, usually in Word-

[1] https://simple.wikipedia.org



*Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)*, pages 63–68, Beijing, China, July 26-31, 2015. 

Net, for a predefined set of complex words (Carroll et al., 1998; Bautista et al., 2009), and then choosing the "simplest" of these synonyms, typically using some frequency-based (Devlin and Tait, 1998; De Belder and Moens, 2010) or length-based heuristics (Bautista et al., 2009). The main shortcomings of the rule-based systems include low recall (De Belder and Moens, 2010) and misclassification of simple words as complex (and vice versa) (Shardlow, 2014).

The paradigm shift from knowledge-based to data-driven simplification came with the creation of Simple Wikipedia, which, aligned with the "original" Wikipedia, constitutes a large comparable corpus to learn from. Yatskar et al. (2010) used the edit history of Simple Wikipedia to recognize lexical simplifications. They employed a probabilistic model to discern simplification edits from other types of content changes. Biran et al. (2011) presented an unsupervised method for learning substitution pairs from a corpus of comparable texts from Wikipedia and Simple Wikipedia, although they exploited the (co-)occurrence statistics of the simplified corpora rather than its metadata. Horn et al. (2014) proposed a supervised framework for learning simplification rules. Using a sentence-aligned simplified corpus, they generated the candidate rules for lexical simplification. A context-aware binary classifier, trained and evaluated on 500 Wikipedia sentences (annotated via crowdsourcing), then decides whether a candidate rule should be applied or not in a certain context.

The main limitation of the aforementioned methods is the dependence on simplified corpora and WordNet. In contrast, we propose a resource-light approach to lexical simplification that requires only a sufficiently large corpus of regular text, making it applicable to the many languages lacking these resources.

## 3 Resource-Light Lexical Simplification

At the core of our lexical simplification method, which we name LIGHT-LS, is the observation that "simple" words, besides being frequent in simplified text, are also present in abundance in regular text. This would mean that we can find simpler synonyms of complex words in regular corpora, provided that reliable methods for measuring (1) the "complexity" of the word and (2) semantic similarity of words are available. LIGHT-LS simplifies only single words, but we fully account for this in the evaluation, i.e., LIGHT-LS is penalised for not simplifying multi-word expressions. In this work, we associate word complexity with the commonness of the word in the corpus, and not with the length of the word.

### 3.1 Simplification Candidate Selection

We employ GloVe (Pennington et al., 2014), a state-of-the-art model of distributional lexical semantics to obtain vector representations for all corpus words. The semantic similarity of two words is computed as the cosine of the angle between their corresponding GloVe vectors. For each content word (noun, verb, adjective, or adverb) $w$, we select as simplification candidates the top $n$ words whose GloVe vectors are most similar to that of word $w$. In all experiments, we used 200-dimensional GloVe vectors pretrained on the merge of the English Wikipedia and Gigaword 5 corpus.[2] For each content word $w$, we select $n = 10$ most similar candidate words, excluding the morphological derivations of $w$.

### 3.2 Goodness-of-Simplification Features

We rank the simplification candidates according to several features. Each of the features captures one aspect of the suitability of the candidate word to replace the original word. The following are the descriptions for each of the features.

**Semantic similarity.** This feature is computed as the cosine of the angle between the GloVe vector of the original word and the GloVe vector of the simplification candidate.

**Context similarity.** Since type-based distributional lexico-semantic models do not discern senses of polysemous words, considering only semantic similarity between the original and candidate word may lead to choosing a synonym of the wrong sense as simplification of the complex word. The simplification candidates that are synonyms of the correct sense of the original word should be more semantically similar to the context of the original word. Therefore, we compute this feature by averaging the semantic similarities of the simplification candidate and each content word from the context of the original word:

$$csim(w, c) = \frac{1}{|C(w)|} \sum_{w' \in C(w)} \cos(\mathbf{v_w}, \mathbf{v_{w'}})$$

[2] http://www-nlp.stanford.edu/data/glove.6B.200d.txt.gz

where $C(w)$ is the set of context words of the original word $w$ and $\mathbf{v_w}$ is the GloVe vector of the word $w$. We use as context a symmetric window of size three around the content word.

**Difference of information contents.** The primary purpose of this feature is to determine whether the simplification candidate is more informative than the original word. Under the hypothesis that the word's informativeness correlates with its complexity (Devlin and Unthank, 2006), we choose the candidate which is less informative than the original word. The complexity of the word is estimated by its information content (*ic*), computed as follows:

$$ic(w) = -\log \frac{freq(w)+1}{\sum_{w' \in C} freq(w')+1}$$

where $freq(w)$ is the frequency of the word $w$ in a large corpus $C$, which, in our case, was the Google Book Ngrams corpus (Michel et al., 2011). The final feature value is the difference between the information contents of the original word and the simplification candidate, approximating the complexity reduction (or gain) that would be introduced should the simplification candidate replace the original word.

**Language model features.** The rationale for having language model features is obvious – a simplification candidate is more likely to be a compatible substitute if it fits into the sequence of words preceding and following the original word. Let $w_{-2}w_{-1}ww_1w_2$ be the context of the original word $w$. We consider a simplification candidate $c$ to be a good substitute for $w$ if $w_{-2}w_{-1}cw_1w_2$ is a likely sequence according to the language model. We employed the Berkeley language model (Pauls and Klein, 2011) to compute the likelihoods. Since Berkeley LM contains only bigrams and trigrams, we retrieve the likelihoods for ngrams $w_{-1}c$, $cw_1$, $w_{-2}w_{-1}c$, $cw_1w_2$, and $w_{-1}cw_1$, for each simplification candidate $c$.

### 3.3 Simplification Algorithm

The overall simplification algorithm is given in Algorithm 1. Upon retrieving the simplification candidates for each content word (line 4), we compute each of the features for each of the simplification candidates (lines 5–8) and rank the candidates according to feature scores (line 9). We choose as the best candidate the one with the highest average rank over all features (line 12). One important thing to notice is, that even though LIGHT-LS has no dedicated component for deciding whether simplifying a word is necessary, it accounts for this implicitly by performing the simplification only if the best candidate has lower information content than the original word (lines 13–15). Since simplification candidates need not have the same POS tag as the original word, to preserve grammaticality, we transform the chosen candidate into the morphological form that matches the POS-tag of the original word (line 14) using the NodeBox Linguistics tool.[3]

**Algorithm 1:** Simplify($tt$)

```
1:  subst ← ∅
2:  for each content token t ∈ tt do
3:      all_ranks ← ∅
4:      scs ← most_similar(t)
5:      for each feature f do
6:      scores ← ∅
7:          for each sc ∈ scs do
8:              scores ← scores ∪ f(sc)
9:          rank ← rank_numbers(scores)
10:         all_ranks ← all_ranks ∪ rank
11:     avg_rank ← average(all_ranks)
12:     best ← argmax_sc(avg_rank)
13:     if ic(best) < ic(tt) do
14:         bpos ← in_pos(best, pos(tt))
15:         subst ← subst ∪ (tt, bpos)
16: return subst
```

## 4 Evaluation

We evaluate the effectiveness of LIGHT-LS automatically on two different datasets but we also let humans judge the quality of LIGHT-LS's simplifications.

### 4.1 Replacement Task

We first evaluated LIGHT-LS on the dataset crowdsourced by Horn et al. (2014) where manual simplifications for each target word were collected from 50 people. We used the same three evaluation metrics as Horn et al. (2014): (1) *precision* is the percentage of correct simplifications (i.e., the system simplification was found in the list of manual simplifications) out of all the simplifications made by the system; (2) *changed* is the percentage of target words changed by the system; and (3) *accuracy* is the percentage of correct simplifications out of all words that should have been simplified.

[3] https://www.nodebox.net

Table 1: Performance on the replacement task

| Model | Precision | Accuracy | Changed |
|---|---|---|---|
| Biran et al. (2011) | 71.4 | 3.4 | 5.2 |
| Horn et al. (2014) | **76.1** | 66.3 | 86.3 |
| LIGHT-LS | 71.0 | **68.2** | **96.0** |

LIGHT-LS's performance on this dataset is shown in Table 1 along with the performance of the supervised system by Horn et al. (2014) and the unsupervised system by Biran et al. (2011), which both used simplified corpora. The results show that LIGHT-LS significantly outperforms the unsupervised system of Biran et al. (2011) and performs comparably to the supervised system of Horn et al. (2014), which requires sentence-aligned simplified corpora. The unsupervised system of Biran et al. (2011) achieves precision similar to that of LIGHT-LS but at the cost of changing only about 5% of complex words, which results in very low accuracy. Our method numerically outperforms the supervised method of Horn et al. (2014), but the difference is not statistically significant.

### 4.2 Ranking Task

We next evaluated LIGHT-LS on the SemEval-2012 lexical simplification task for English (Specia et al., 2012), which focused on ranking a target word (in a context) and three candidate replacements, from the simplest to the most complex. To account for the peculiarity of the task where the target word is also one of the simplification candidates, we modified the features as follows (otherwise, an unfair advantage would be given to the target word): (1) we excluded the *semantic similarity* feature, and (2) we used the information content of the candidate instead of the difference of information contents.

We used the official SemEval task evaluation script to compute the Cohen's kappa index for the agreement on the ordering for each pair of candidates. The performance of LIGHT-LS together with results of the best-performing system (Jauhar and Specia, 2012) from the SemEval-2012 task and two baselines (random and frequency-based) is given in Table 2. LIGHT-LS significantly outperforms the supervised model by Jauhar and Specia (2012) with $p < 0.05$, according to the non-parametric stratified shuffling test (Yeh, 2000). An interesting observation is that the competitive frequency-based baseline highly correlates with our information content-based feature (the higher the frequency, the lower the information content).

Table 2: SemEval-2012 Task 1 performance

| Model | $\kappa$ |
|---|---|
| baseline-random | 0.013 |
| baseline-frequency | 0.471 |
| Jauhar and Specia (2012) | 0.496 |
| LIGHT-LS | **0.540** |

### 4.3 Human Evaluation

Although automated task-specific evaluations provide useful indications of a method's performance, they are not as reliable as human assessment of simplification quality. In line with previous work (Woodsend and Lapata, 2011; Wubben et al., 2012), we let human evaluators judge the grammaticality, simplicity, and meaning preservation of the simplified text. We compiled a dataset of 80 sentence-aligned pairs from Wikipedia and Simple Wikipedia and simplified the original sentences with LIGHT-LS and the publicly available system of Biran et al. (2011). We then let two annotators (with prior experience in simplification annotations) grade grammaticality and simplicity for the manual simplification from Simple Wikipedia and simplifications produced by each of the two systems (total of 320 annotations per annotator). We also paired the original sentence with each of the three simplifications (manual and two systems') and let annotators grade how well the simplification preserves the meaning of the original sentence (total of 240 annotations per annotator). We averaged the grades of the two annotators for the final evaluation. All grades were assigned on a Likert (1–5) scale, with 5 being the highest grade, i.e., all fives indicate a very simple and completely grammatical sentence which fully preserves the meaning of the original text. The inter-annotator agreement, measured by Pearson correlation coefficient, was the highest for grammaticality (0.71), followed by meaning preservation (0.62) and simplicity (0.57), which we consider to be a fair agreement, especially for inherently subjective notions of simplicity and meaning preservation.

The results of human evaluation are shown in Table 3. In addition to grammaticality (Gr), simplicity (Smp), and meaning preservation (MP), we measured the percentage of sentences with at least one change made by the system (Ch). The results imply that the sentences produced by LIGHT-

Table 4: Example simplifications

| Source | Sentence |
|---|---|
| Original sentence | *The contrast between a high level of education and a low level of political rights was particularly great in Aarau, and the city refused to send troops to defend the Bernese border.* |
| Biran et al. (2011) simpl. | *The **separate** between a high level of education and a low level of political rights was particularly great in Aarau , and the city refused to send troops to defend the Bernese border.* |
| LIGHT-LS simpl. | *The contrast between a high level of education and a low level of political rights was **especially** great in Aarau, and the city **asked** to send troops to **protect** the Bernese border.* |

Table 3: Human evaluation results

| Source | Gr | Smp | MP | Ch |
|---|---|---|---|---|
| Original sentence | 4.90 | 3.36 | – | – |
| Manual simplification | 4.83 | 3.95 | 4.71 | 76.3% |
| Biran et al. (2011) | **4.63** | 3.24 | **4.65** | 17.5% |
| LIGHT-LS | 4.60 | **3.76** | 4.13 | **68.6%** |
| Biran et al. (2011) Ch. | 3.97 | 2.86 | 3.57 | – |
| LIGHT-LS Ch. | 4.57 | 3.55 | 3.75 | – |

LS are significantly simpler ($p < 0.01$; paired Student's t-test) than both the original sentences and sentences produced by the system of Biran et al. (2011). The system of Biran et al. (2011) produces sentences which preserve meaning better than the sentences produced by LIGHT-LS, but this is merely because their system performs no simplifications in over 80% of sentences, which is something that we have already observed on the replacement task evaluation. Furthermore, annotators found the sentences produced by this system to be more complex than the original sentences. On the contrary, LIGHT-LS simplifies almost 70% of sentences, producing significantly simpler text while preserving grammaticality and, to a large extent, the original meaning.

In order to allow for a more revealing comparison of the two systems, we additionally evaluated each of the systems only on sentences on which they proposed at least one simplification (in 70% of sentences for LIGHT-LS and in only 17.5% of sentences for the system of Biran et al. (2011)). These results, shown in the last two rows of Table 3, demonstrate that, besides simplicity and grammaticality, LIGHT-LS also performs better in terms of meaning preservation. In Table 4 we show the output of both systems for one of the few example sentences in which both systems made at least one change.

Since LIGHT-LS obtained the lowest average grade for meaning preservation, we looked deeper into the causes of changes in meaning introduced by LIGHT-LS. Most changes in meaning stem from the inability to discern synonymy from relatedness (or even antonymy) using GloVe vectors. For example, the word "cool" was the best simplification candidate found by LIGHT-LS for the target word *"warm"* in the sentence *"Water temperatures remained warm enough for development"*.

## 5 Conclusion

We presented LIGHT-LS, a novel unsupervised approach to lexical simplification that, unlike existing methods, does not rely on Simple Wikipedia and lexicons like WordNet, which makes it applicable in settings where such resources are not available. With the state-of-the-art word vector representations at its core, LIGHT-LS requires nothing but a large regular corpus to perform lexical simplifications.

Three different evaluation settings have shown that LIGHT-LS's simplifications based on multiple features (e.g., information content reduction, contextual similarity) computed on regular corpora lead to performance comparable to that of systems using lexicons and simplified corpora.

At the moment, LIGHT-LS supports only single-word simplifications but we plan to extend it to support multi-word expressions. Other lines of future research will focus on binding LIGHT-LS with methods for syntax-based (Zhu et al., 2010) and content-based (Glavaš and Štajner, 2013) text simplification.

## Acknowledgements

This work has been partially supported by the Ministry of Science, Education and Sports, Republic of Croatia under the Grant 036-1300646-1986. We thank the anonymous reviewers for their useful comments.

## References


Susana Bautista, Pablo Gervás, and R. Ignacio Madrid. 2009. Feasibility analysis for semi-automatic conversion of text to improve readability. In *Proceedings of the Second International Conference on Information and Communication Technology and Accessibility (ICTA)*, pages 33–40.

Or Biran, Samuel Brody, and Noémie Elhadad. 2011. Putting it simply: A context-aware approach to lexical simplification. In *Proceedings of the ACL-HLT 2011*, pages 496–501. ACL.

John Carroll, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait. 1998. Practical simplification of english newspaper text to assist aphasic readers. In *Proceedings of AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*, pages 7–10.

Jan De Belder and Marie-Francine Moens. 2010. Text simplification for children. In *Proceedings of the SIGIR Workshop on Accessible Search Systems*, pages 19–26.

Siobhan Devlin and John Tait. 1998. The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic Databases*, pages 161–173.

Siobhan Devlin and Gary Unthank. 2006. Helping aphasic people process online information. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*, pages 225–226. ACM.

Christiane Fellbaum. 1998. *WordNet*. Wiley Online Library.

Lijun Feng. 2009. Automatic readability assessment for people with intellectual disabilities. In *ACM SIGACCESS Accessibility and Computing*, number 93, pages 84–91. ACM.

Goran Glavaš and Sanja Štajner. 2013. Event-centered simplification of news stories. In *Proceedings of the Student Workshop held in conjunction with RANLP*, pages 71–78.

Colby Horn, Cathryn Manduca, and David Kauchak. 2014. Learning a lexical simplifier using wikipedia. In *Proceedings of ACL 2014 (Short Papers)*, pages 458–463.

Sujay Kumar Jauhar and Lucia Specia. 2012. UOW-SHEF: SimpLex – lexical simplicity ranking based on contextual and psycholinguistic features. In *Proceedings of the SemEval-2012*, pages 477–481. ACL.

Juan Martos, Sandra Freire, Ana González, David Gil, and Maria Sebastian. 2012. D2.1: Functional requirements specifications and user preference survey. Technical report, FIRST project.

Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, and Jon Orwant. 2011. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182.

Adam Pauls and Dan Klein. 2011. Faster and smaller n-gram language models. In *Proceedings of ACL-HLT 2011*, pages 258–267. ACL.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of EMNLP 2014*, pages 1532–1543.

Sarah E. Petersen and Mari Ostendorf. 2007. Text simplification for language learners: A corpus analysis. In *Proceedings of Workshop on Speech and Language Technology for Education (SLaTE)*.

Luz Rello. 2012. DysWebxia: A Model to Improve Accessibility of the Textual Web for Dyslexic Users. In *ACM SIGACCESS Accessibility and Computing.*, number 102, pages 41–44. ACM, New York, NY, USA, January.

Horacio Saggion, Sanja Štajner, Stefan Bott, Simon Mille, Luz Rello, and Biljana Drndarevic. 2015. Making it simplext: Implementation and evaluation of a text simplification system for spanish. *ACM Transactions on Accessible Computing*, 6(4):14.

Matthew Shardlow. 2014. Out in the open: Finding and categorising errors in the lexical simplification pipeline. In *Proceedings of LREC 2014*, pages 1583–1590.

Lucia Specia, Sujay Kumar Jauhar, and Rada Mihalcea. 2012. SemEval-2012 Task 1: English lexical simplification. In *Proceedings of the SemEval 2012*, pages 347–355. ACL.

Kristian Woodsend and Mirella Lapata. 2011. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of EMNLP 2011*, pages 409–420. ACL.

Sander Wubben, Antal Van Den Bosch, and Emiel Krahmer. 2012. Sentence simplification by monolingual machine translation. In *Proceedings of ACL 2012 (Long Papers)*, pages 1015–1024. ACL.

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 *Proceedings of NAACL 2010*, pages 365–368. ACL.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of COLING 2000*, pages 947–953. ACL.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the COLING 2010*, pages 1353–1361. ACL.