

As Simple As It Gets - A sentence simplifier for different learning levels and contexts

Abstract—This paper presents a text simplification method that transforms complex sentences into simplified forms. Our method uses NLP-techniques to simplify the text based on the target audience context, improving its overall understandability. We evaluate our approach in two aspects: grammatical structure and understandability. In both aspects, our approach achieved good results, showing its applicability to the learning process.

I. INTRODUCTION

Reading is an integral part of any learning process. The rapid expansion of information and knowledge, in particular, available on the Web, requires continuous learning and knowledge acquisition, where reading is a substantial activity. Reading for learning often comprises writing actions (annotations), the so-called active reading [1]. In most cases, these annotations aim at reinforcing the understanding of the text, which often comes attached to a segment that needs more contemplation due to its importance or complexity [8], [11]. There are many techniques that aim at reducing the cognitive overhead of reading activities. In particular, one common practice is the application of *text simplification*.

Text simplification (or sentence simplification) describes the process of producing a simplified version of a text which preserves its original semantic meaning [3], [4]. It can be achieved by different strategies, for example, by changing the grammatical structure of the sentence or by lexical replacement. The benefits of text simplification can affect many readers, in particular language learners [12], people with reading disabilities [7] such as aphasia [3], and low-literacy readers [14].

As an extension of previous tools, this work enables the creation of simplified versions of text focused on a specific context. Our approach aims at simplifying texts by transforming the sentences into simpler and more understandable statements which use most common and popular terminology. The goal is to adapt the text to a specific target audience and to a specific learning context. This allows people from different learning levels or learning backgrounds to consume contextualized/personalized versions of a text/book. Our approach consists of (i) lexical annotation of sentences, (ii) identification of most suitable synonyms, (iii) generation of context-based content, and finally (iv) validation of the generated sentences.

II. PROBLEM DEFINITION

Briefly, we define a sentence as a list of words $O = \langle w_1, w_2, \dots, w_l \rangle$; we say that w_i occurs in O . We also

consider a function S that assigns a set synonyms $S(w_i)$ to each word w_i , and a part-of-speech tagging function p that assigns a part-of-speech tag, or briefly a POS, $p(u)$ to each word or synonym u ; the function p is such that $p(w_i) = p(u_{ij})$, for each word w_i and each synonym u_{ij} in $S(w_i)$.

Since all synonyms of a word have the same POS as the word, the set of synonyms is filtered by sense through a word sense disambiguation step. Thus, we introduce a function δ_{sense} that assigns a word sense $\delta_{sense}(w_i, O)$ to each word w_i that occurs in a sentence O . We extend δ_{sense} to the synonyms of a word w_i so that $\delta_{sense}(w_i, O) = \delta_{sense}(u_{ij}, O)$, for each synonym u_{ij} in $S(w_i)$. The resulting simplified sentence R is synthesized using the most common word in a given context.

Finally, our approach iteratively validates the lexical replacements comparing the popularity of a subset $\{w_i, \dots, w_{i+n}\}$ of n consecutive words of R , starting on the i^{th} word, to a subset $\{r_i, \dots, r_{i+n}\}$ of n consecutive words of O , also starting on the i^{th} word, where $i + n \leq |O|$, and $|T| = |O|$. The popularity function $\phi_{popularity}$ assigns the number of occurrences of a subset of words in a given context. Intuitively, we consider a subset of size n that runs over the lists of words O and R , such as a sliding window algorithm. The sliding window checks if $\phi_{popularity}(\{w_i, \dots, w_{i+n}\}) > \phi_{popularity}(\{r_i, \dots, r_{i+n}\})$. The sets of words that are more popular in O than in R are replaced by the original one, since they are considered simpler than the candidate replacements.

III. METHOD

In this section, we present our method for sentence simplification depicted in Figure 1. The method is divided into 4 main steps: (i) part-of-speech (POS) tagging; (ii) synonym probing; (iii) context frequency-based lexical replacement; and (iv) sentence checker.

A. Part-of-speech tagging

POS tagging is a fundamental step for the task of sentence simplification. Since a word can have multiple POSs, determining the correct POS helps us find the most suitable synonyms for a given word in a particular context.

For instance, the word “love” can be tagged as a *noun* or a *verb*, and the word “narrative” can be tagged as a *noun* or an *adjective* (as in “narrative poetry”) in a given context. Thus, depending on the context, we will determine the right POS tagging for a word.

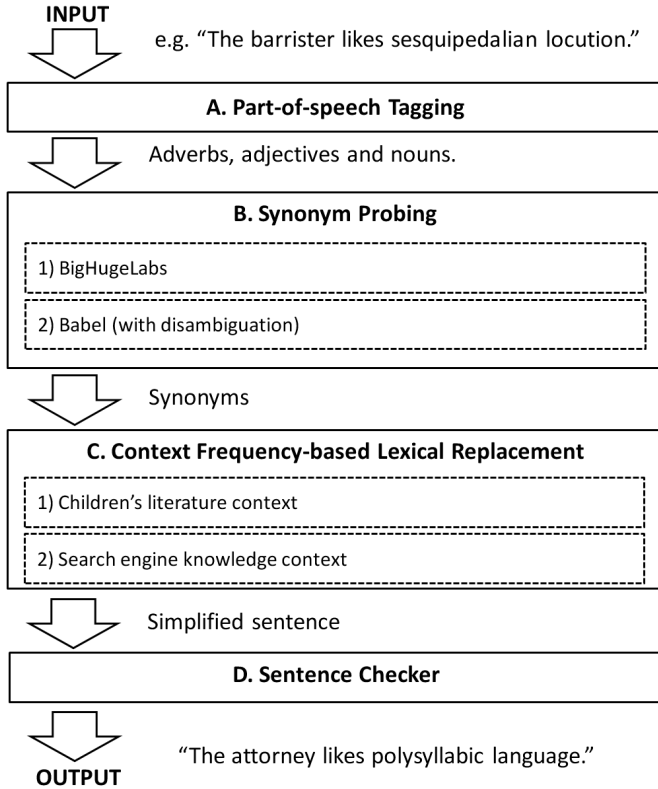


Figure 1. Simplification workflow.

Let a sentence O be represented by the list of words $\langle 'I', 'read', 'a', 'love', 'narrative' \rangle$, then the function $p('love')$ returns the POS tag *noun*. In this context, “love” is a noun acting as an adjective that describes the type of the narrative, which is also a noun.

Hence, with the lexical information, we prevent replacement of words that belong to different lexical categories. In the example above, the noun “love” must not be replaced by a verb, because it would lead to (a) a grammatical flaw or (b) a different sense of a word. We will approach (b) in the next steps. Thus, although “enjoy” might be a synonym for “love”, the word “enjoy” is a potential synonym of the verb “love”, while “passion” would be a potential synonym of the noun “love”.

In order to recognize the lexical items and prevent grammatical flaws, we used a state-of-art tool, Stanford Log-linear Part-Of-Speech Tagger [13]. This tool is based on the Penn Treebank Tagset [9], which describes 36 POS taggers. Our work focuses on 3 groups (*adjectives*, *nouns* and *adverbs*) that cover 10 types of tags in the Penn tagset¹. Thus, given any sentence as input, the first step is responsible for annotating and outputting the POS-tagged sentence.

¹http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

B. Synonym probing

In this step, we identify synonyms of given nouns, adverbs and adjectives of a sentence. After processing a sentence to be simplified (Section III-A), a set of synonyms $S(w_i)$ is assigned for each word w_i according to its part-of-speech. Thus, for each synonym u_{ij} in $S(w_i)$, $p(w_i) = p(u_{ij})$.

Following the example in the Section III-A, a set of synonyms for “love” could be “passion”, “beloved” or “dear”, while for “narrative” could be “story”, “narration” or “tale”.

However, inspecting the set of synonyms found for “love”, it is clear that a random substitution of a word for a synonym might change the sense of the sentence. Thus, to express similar or equivalent meaning of a word w_i in a sentence O , the set of synonyms $S(w_i)$ retrieved from a thesaurus is filtered by sense, $\delta_{sense}(w_i, O) = \delta_{sense}(u_{ij}, O)$.

Navigli and Ponzetto [10] developed the Babelnet API², which uses WordNet³ to identify the sense of a word in a certain context.

WordNet is the biggest lexical database in English, where a word (adjectives, adverbs, nouns or verbs) is grouped with other words that denote the same concept (also known as synsets - sets of cognitive synonyms). Thus, through Babelnet API, for each word in a sentence, a semantic graph is generated. Exploiting the word relations in this graph, we determine the right synset for a word and the correct contextualized synonyms. Finally, the set of synonyms is filtered and this step outputs the word and its synonyms in a specific context.

In addition, we used a thesaurus database⁴. Note that this thesaurus does not provide the sense of each word. Thus, in this case, we only matched the lexical categories, $p(w_i) = p(u_{ij})$, i.e., *noun to noun*, *adverbs to adverbs* and so on.

C. Context frequency-based lexical replacement

After the set of synonyms is retrieved and filtered by sense, the next step aims at identifying the synonym for a word that best fits in a determined context. Thus, we need to identify which lexical replacement is the best choice to maximize the understandability of the input sentence. For this, we rely on the assumption that the most frequently occurring word in a controlled vocabulary (extracted from a specific domain) is of tacit knowledge. From now on, we call this assumption *word popularity*.

For instance, in our previous example, “passion” is the only synonym found for the noun “love”, while “story”, “tale” amongst others, are synonyms for “narrative”. However, as the word popularity of “love” is greater than “passion”, the word “love” is kept, but in the second case, the word popularity of “story” is greater than “narrative”, resulting in the sentence “love story”. Indeed, this is the most common formulation.

²<http://lcl.uniroma1.it/babelnet/>

³<http://wordnet.princeton.edu/>

⁴<http://words.bighugelabs.com>

In this manner, we can focus on a specific domain to simplify a sentence according to a target audience. Given a controlled vocabulary, our method is able to select the most suitable words that match a context level. To illustrate this, we describe two contexts: (1) children’s literature context, and (2) search engine knowledge context.

1) *Children’s literature context*: The goal of using children’s literature is to simplify the sentences to a level that they become understandable to young kids. Thus, to build this context, we crawled several books written for kids between 5 and 8 years old and measured the number and the frequency of words. In total, it resulted in a dictionary with 2537 distinct words. It is noteworthy that the number of new words converged after the 20th book crawled.

As a result, we are able to detect which of the synonyms is the most common in the children’s context. In our example, “story” is far more popular than “narrative” (in fact, “narrative” is not even included in this contextualized vocabulary). Hence, we assume that, if a word is popular in a given context, then the word is known by its audience, in this case, by children.

2) *Search engine knowledge context*: Search engines crawl content available on the Web. Hence, they have an inherent knowledge that can be exploited to obtain the most common words used in a given language. Given the fact that Web pages are generated by humans, results of search engines implicitly represent the common sense.

We used this information to help in the task of finding popular words. Given a set of candidate synonyms, we query them using a search engine to retrieve the number of pages that contains each word. The higher the number of Web pages containing a given word, the more popular it is and the higher is the probability of a person to know it.

Our method uses the Yahoo API⁵ to retrieve the number of pages that contains a word.

D. Sentence checker

Following the same strategy of Section III-C2, we use the search engine knowledge to check if a given sentence occurs on a high scale on the Web. The main goal of this step is to validate the new sentence structure.

Although a synonym may be *simpler* than another, it may happen that it is rarely used in the context of a sentence. Thus, given the output of the previous step, we once again query the search engine with split sentences in order to identify the most common arrangement.

We extend the assumption of word popularity to *n-gram popularity*, where *n* is at most the total number of words in a sentence. If *n* is lower than the number of words in a sentence, the algorithm to validate the lexical replacements works as a sliding window algorithm.

⁵<http://developer.yahoo.com>

Given $O = \langle w_1, w_2, \dots, w_l \rangle$, $S = \langle s_1, s_2, \dots, s_l \rangle$ and $R = \langle r_1, r_2, \dots, r_l \rangle$, where O represents the original sentence, s_i represents the most popular synonyms of each word w_i in O , and R is the resulting simplified sentence for $i = 1, \dots, l$, and thus, the lexical replacements made during the simplification process are checked according to the search engine knowledge. We query a set of words in O and R and keep the most popular set of words. The set of words are queried as a sliding window algorithm, where, once the size n of the window is set, each subset of words are selected to replace the original set of words in O .

IV. EVALUATION PROCESS

Our evaluation is divided into two steps. The first part of the evaluation aims at validating the method with respect to preservation of the original meaning and its grammatical correctness. The second part of the evaluation aims at measuring improvement in the understandability for the reader, given the original sentence and its simplified form.

A. Evaluation 1 - Preservation of Meaning and Correctness

Focusing on the native English speaker, our main goal is to validate our simplification process regarding potential errors introduced by our method and if the texts preserve the original meaning. Thus, in this evaluation, we present to the participant a text retrieved from our dataset and its simplified form. The questionnaire for the native English speakers is:

- 1) Do the texts above have the same meaning? (yes/no)
- 2) Are the text free from grammar errors? (yes/no)

B. Evaluation 2 - Simplification

After the feedback from native English speakers, we selected the texts that were marked as free from grammar errors and that had the same meaning. Hence, the second evaluation with the non-native English speakers is focused on the main goal of our approach, i.e., to validate if our simplification method improves the understandability for English-speaking learners. As the sentences of the dataset are generally easy to understand and the English level of the participants are different from each other, they could select between the original sentence, simplified or say that it is indifferent.

The questionnaire presented for the non-native English speakers is composed by the following simple question:

- 1) Which sentence is easier to understand? (original/simplified/indifferent)

C. Dataset

As for the dataset, we crawled random snippets from 20 books. In total we gathered 1261 sentences to be simplified using the methods described in Section III. For each book, we tokenized the sentences using the Stanford NLP tool to keep the sentence structure.

Table I
RESULTS OF THE SIMPLIFICATION SENTENCE METHOD FOR DIFFERENT STRATEGIES (PARAMETER SETTINGS) FROM THE EVALUATION WITH NATIVE ENGLISH SPEAKERS.

Strategy ID	Window's size	Vocabulary source	Synonym source	Precision (same meaning)	Precision (grammatically correct)
S_1	1	Children's literature	WordNet	81%	67%
S_2	1	Children's literature	BigHugeLabs	56%	48%
S_3	2	Search Engine	WordNet	80%	66%
S_4	2	Search Engine	BigHugeLabs	55%	55%
S_5	3	Search Engine	WordNet	82%	61%
S_6	3	Search Engine	BigHugeLabs	61%	59%
S_7	1	Search Engine	WordNet	81%	51%
S_8	1	Search Engine	BigHugeLabs	52%	60%

D. Evaluation setup

As described in Section III, the simplification tool contains many parameters for each of which the settings must be specified. Here, we describe the parameters for setting the synonym source, the controlled vocabulary and the windows size of the sentence checker. Our goal is to provide a tool that can be adapted to a specific context. In this manner, the following 3 parameters must be defined: (1) synonym source, (2) controlled vocabulary and (3) windows size.

1) *Synonym source*: This parameter is used to control the synonyms suggested for a given word. In our experiments we used WordNet and BigHugeLabs (described in III-B).

2) *Controlled vocabulary*: This parameter is used to customize the simplification to a target audience. Although the list of synonyms provides words with the same sense, a specific word might not be used by a target audience, thus the controlled vocabulary will assist in picking up the right synonym in a given context. We used two vocabularies, one extracted from children's books and another from search engines (see III-C for more details).

3) *Window sizes*: This parameter defines the boundaries of a sentence. The set of words will be checked regarding its popularity, i.e., to prevent obscure and rare sentence formulations. We set the windows sizes between 1 and 3.

V. RESULTS

This section presents the results for the two evaluations and different parameter settings described in Section IV.

The first questionnaire was answered by 77 native English speakers and covered all sentences in the dataset (original and simplified sentences), while the second questionnaire was answered by 19 non-native English speakers and covered almost 50% of the total amount of sentences in the dataset.

Table I presents the results of the evaluations with the native English speakers. The column "Precision (same meaning)" shows the agreement of the evaluators regarding the sense similarity between the original and the simplified sentence; the column "Precision (grammatically correct)"

Table II
RESULTS OBTAINED FROM THE EVALUATION WITH NON-NATIVE ENGLISH SPEAKERS FOR DIFFERENT STRATEGIES (PARAMETER SETTINGS).

Strategy ID	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
Simplified	34%	30%	38%	41%	34%	28%	28%	21%

shows the rate of the sentences that were simplified and were free from grammatical errors.

The results are also discriminated regarding their different configuration settings which we vary the window's size, the controlled vocabulary and the synonyms source.

As for non-native English speakers, Table II shows the percentage of cases where the simplified version was easier to understand. In none of the cases, the original sentence was selected. The complementary percentages are all allocated to the choice *indifferent*.

VI. RELATED WORK

Paraphrasing has always been used as the main instrument for clarifying and simplifying sentences. It supports readers to better understand the original content in many scenarios, for example, readers that are trying to understand a complex text, language learners and even readers with disability (such as aphasia).

Aiming at making newspaper text accessible to aphasics Carroll et al. [3] and Canning and Tait [2] proposed the application of syntactical and lexical simplification. Syntactical simplification, for example, constitutes replacing passive constructions with active ones, eliminating multiple embedded prepositional and relative phrases replacing longer sentences with two or more short ones. As presented in this paper, we focus on the lexical simplification, which consists in simplifying word by word or a set of words [6].

In order to validate a data-driven approach to the sentence simplification task, Zhu et al. [15] used paired documents in English Wikipedia and Simple Wikipedia. Their tree-based translation model for sentence simplification covers splitting, dropping, reordering and word/phrase substitution. However, their "Word Substitution" schema is rather superficial, based solely on word probability. The authors do not provide any

information on the dictionaries used and there is no analysis on the effects of out of context replacements. Using a similar approach, Coster and Kauchak[5] exploit a parallel corpus of paired documents from English Wikipedia and Simple Wikipedia to train a phrase-based machine translation model. Unfortunately, none of them perform user studies to validate the results with real subjects.

As described in this paper, we follow the same goals of the related work presented above but with a deepen strategy focused on lexical replacement in a specific context. We proposed a monolingual machine translation technique where the output should be simpler than the input sentence but similar in meaning. Furthermore, we validate our method with real human subjects.

VII. CONCLUSION AND FUTURE DIRECTIONS

In this paper, we presented a sentence simplification method that aims at improving the understandability of given phrases, in our case, in the English language. The simplified versions produced by our method can assist language learners, people with reading disabilities and general learners with different background levels. Our approach demonstrated its usefulness in the adaptation of contents in different contexts - children's literature and search engine knowledge context, which represents the general public's knowledge.

The results of our user studies showed that in the children's context and search engine knowledge context the text simplification preserved the original meaning in approximately 80% while almost 70% of the texts were grammatically correct. Additionally, in almost 40% of the cases, the simplified versions of the sentences were easier to understand, while the remaining sentences were indifferent, regarding its comprehensibility. As for future work, we plan to achieve better precision of the simplification and to eliminate grammatical errors and misunderstandings. Additionally, we plan to include the simplifications of verbs (the challenge is to identify the right conjugation) and finally build contextualized simplified vocabularies for different learning branches.

REFERENCES

- [1] M. J. Adler and C. V. Doren. *How to Read a Book*. Revised edition, Simon and Schuster, New York, 1972.
- [2] Y. Canning and J. Tait. Syntactic simplification of newspaper text for aphasic readers. In *Proceedings of SIGIR-99 Workshop on Customised Information Delivery*, pages 6–11, 1999.
- [3] J. Carroll, G. Minnen, Y. Canning, S. Devlin, and J. Tait. Practical simplification of english newspaper text to assist aphasic readers. In *In Proc. of AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*, pages 7–10, 1998.
- [4] R. Chandrasekar and B. Srinivas. Automatic induction of rules for text simplification, 1997.
- [5] W. Coster and D. Kauchak. Learning to simplify sentences using wikipedia. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 1–9, Portland, Oregon, June 2011. Association for Computational Linguistics.
- [6] S. Devlin and J. Tait. The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic Databases*, 1998.
- [7] K. Inui, A. Fujita, T. Takahashi, R. Iida, and T. Iwakura. Text simplification for reading assistance: a project note. In *Proceedings of the second international workshop on Paraphrasing - Volume 16, PARAPHRASE '03*, pages 9–16, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
- [8] R. Kawase, E. Herder, and W. Nejdl. A comparison of paper-based and online annotations in the workplace. In *Proceedings of the 4th European Conference on Technology Enhanced Learning: Learning in the Synergy of Multiple Disciplines, EC-TEL '09*, pages 240–253, Berlin, Heidelberg, 2009. Springer-Verlag.
- [9] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini. Building a large annotated corpus of english: the penn treebank. *Comput. Linguist.*, 19(2):313–330, June 1993.
- [10] R. Navigli and S. P. Ponzetto. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193(0):217 – 250, 2012.
- [11] B. P. Nunes, R. Kawase, S. Dietze, G. H. B. de Campos, and W. Nejdl. Annotation tool for enhancing e-learning courses. In *Advances in Web-Based Learning - ICWL 2012 - 11th International Conference, Sinaia, Romania, September 2-4, 2012. Proceedings*, pages 51–60, 2012.
- [12] A. Siddharthan. An architecture for a text simplification system. In *Proceedings of the Language Engineering Conference, LEC '02*, pages 64–, Washington, DC, USA, 2002. IEEE Computer Society.
- [13] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1, NAACL '03*, pages 173–180, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
- [14] W. M. Watanabe, A. C. Junior, V. R. Uzêda, R. P. d. M. Fortes, T. A. S. Pardo, and S. M. Aluísio. Facilita: reading assistance for low-literacy readers. In *Proceedings of the 27th ACM international conference on Design of communication, SIGDOC '09*, pages 29–36, New York, NY, USA, 2009. ACM.
- [15] Z. Zhu, D. Bernhard, and I. Gurevych. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics, COLING '10*, pages 1353–1361, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.