A New Aligned Simple German Corpus

Vanessa Toborek  Moritz Busch  Malte Boßert  Christian Bauckhage
Pascal Welke
University of Bonn
{toborek, welke}@cs.uni-bonn.de
{busch, mbossert}@uni-bonn.de
chrisitan.bauckhage@iais.fraunhofer.de

Abstract

"Leichte Sprache", the German counterpart to Simple English, is a regulated language aiming to facilitate complex written language that would otherwise stay inaccessible to different groups of people. We present a new sentence-aligned monolingual corpus for Simple German – German. It contains multiple document-aligned sources which we have aligned using automatic sentence-alignment methods. We evaluate our alignments based on a manually labelled subset of aligned documents. The quality of our sentence alignments, as measured by F1-score, surpasses previous work. We publish the dataset under CC BY-SA and the accompanying code under MIT license.

1 Introduction

Text in simple language benefits language learners, people with learning difficulties and children that tend to have a hard time understanding original and especially formal texts due to grammar and vocabulary. Text simplification describes the problem of generating a simplified version of a given text while conveying the same matter [28]. This involves the reduction of lexical and syntactic complexity by various operations like deletion, rewording, insertion, and reordering [26]. Accordingly, text simplification can also encompass additional explanations or simplifications for difficult concepts and a structured layout [28].

To make language more inclusive, guidelines for simple versions of languages have been developed. In English, most notably, Ogden [21] introduced “Basic English” (BE). In German there are two prevalent kinds of simple language: “Einfache Sprache” (ES) and “Leichte Sprache” (LS), both roughly translating to easy language [16]. LS has strict rules, which include e.g. the removal of subordinate clauses, the insertion of paragraphs after each sentence and the separation of compound nouns with hyphens. ES is less restrictive and does not have a specific set of rules; instead, translators can work more freely. However, the goal of both approaches is to improve the language’s accessibility.

Since the problem of text simplification can also be defined as a monolingual translation task, the availability of data becomes a prerequisite in order to apply statistical machine learning models to it. In particular, sentence-aligned text constitutes the backbone of neural machine translation. To the best of our knowledge only the work of Klaper, Ebling, and Volk [13] presents a parallel sentence-aligned corpus in German created from publicly available data from the internet. Our work addresses the lack of data for text simplification in German and thus creates an aligned corpus of easy language.
and corresponding German texts. As there is no German equivalent to the Simple English Wikipedia (which provides cross-lingual references between Simple English and English articles), we had to rely on multiple sources offering a small number of articles in German as well as in some simplified version of it. Our corpus consists mostly of sources providing articles in “Leichte Sprache”, but in order to achieve a larger vocabulary size we also incorporated articles from a website in “Einfache Sprache”. In the following we will always talk about Simple German, whenever the distinction between those two simplification forms is not relevant.

Following the description of our dataset and its collection process, we present the results of a comparison of different sentence-alignment methods. Then, we select the best approach and obtain a sentence-aligned dataset that can potentially be extended by crawling further websites. See Figure 1 to see examples from the sentence alignments. Finally, we discuss the limitations of our dataset and future research related to it. We share our code to build the dataset on GitHub. The repository contains a list of urls and scripts to reproduce the dataset by crawling the (archived) websites, parsing the text and aligning the sentences. Additionally, the fully prepared dataset is available upon request.

2 Related Work

There are various classification systems for language with different aims and scopes. In the context of language learning, the European Council has defined the six proficiency levels A1 to C2 based on competencies in reading, writing, speaking and understanding of language [7]. These are mainly intended to evaluate learners, not texts and are applicable to multiple languages. For English, the Lexile³ scale gives scores on reading proficiency, as well as text complexity, but has been criticized as carrying little qualitative meaning [33]. A particularly early attempt at a “simplified”, controlled English language is Basic English [21]. It is a subset of (rules and words of) English and aimed at being easy to learn. However, it does not restrict sentence length, complexity of content, or implicit context. As a result, even “easy” texts, as measured on one of the above scales, may fall short in comprehensibility and accessibility, if one, e.g. assumes certain mental or physical impairments.

We focus on German texts that are aimed at being (more) inclusive to such target groups, which requires specifically designed rules. “Leichte Sprache” (Simple German) is designed for people with cognitive disabilities [16, 17, 30]. “Einfache Sprache” (Plain German) targets the dissemination of expert contents to lay people and is less comprehensible (and hence less inclusive), but at the same time less perceptible and more acceptable to larger audiences [16].

---

1. https://github.com/mlai-bonn/Simple-German-Corpus
2. toborek@cs.uni-bonn.de
3. www.lexile.com
There are some sources on monolingual parallel corpora for different languages. English – simplified English corpora have been created, e.g. from the Simple English Wikipedia (which does not adhere to any fixed simplification standard) [6, 10, 11, 39]. Using aligned articles from Wikipedia has been criticized, as (i) simple Wikipedia contains many complex sentences and (ii) sentence alignments are improbable, as the articles are often independently written [35]. Hence, an alternative corpus of five difficulty levels targeted at children at different reading levels has been proposed [35, 11]. Spanish [4], Danish [14], and Italian [5] corpora exist as well.

When narrowing the research field down to the German language, only a few resources remain. Klapier, Ebling, and Volk [13] crawl five websites that provide a total of 256 parallel German and Simple German articles, spanning various topics. They provide sentence level alignments and thus their result is the most similar dataset to ours that currently exists. They use a sentence alignment algorithm based on dynamic programming with prior paragraph alignment based on bag-of-word cosine similarities [2] and report an F1-score of 0.085 for their alignments on ground truth. Säuberli, Ebling, and Volk [27] introduce two sentence-aligned corpora gathered from the Austrian Press Agency and from capito. Here, the authors align the sentences of the original texts with their corresponding translation in level A1 and B1 of the Common European Framework of Reference for Languages [7]. The resulting simplifications are very different to the simplifications according to the rules of LS. Additionally, Rios et al. [25] extend this dataset by adding articles from a Swiss news outlet which publishes “simplified” summaries alongside its content which, however, do not adhere to any simplification standard. Here, sentence-level alignments are not provided. Battisti et al. [3] compile a corpus for Simple German that mostly consists of unaligned Simple German articles and 378 parallel article pairs but without sentence-alignments. Aumiller and Gertz [1] present an extensive document-aligned corpus by using the German children encyclopedia “Klexikon”. The authors align the documents by choosing corresponding articles from Wikipedia, making it unlikely that specific sentences can be matched. As republishing may lead to legal ramifications, only the “Klexikon” dataset is publicly available. Overall, current German language text simplification datasets are rare, small, usually not publicly available, and typically not focused on inclusive Simple German.

3 Dataset Description

Generally, there is little to no data tailored towards text simplification. Our work addresses the lack of data for simple language in German. Besides the problems of text simplification and automatic accessibility assessment, we had other possible use cases like text summarization and curriculum learning during the creation of the dataset in mind. We will actively maintain the dataset and hope it will be adapted by the community such that over time further web sources will be integrated.

3.1 Composition

While there is a range of different websites regularly publishing news articles in simple language, only a fraction of them maintains parallel versions in German and Simple German. Restricting ourselves to sources providing both versions of an article, we obtained seven different websites. We made sure to only use freely available articles. Table 3 provides an overview of all websites with a brief description of their content.

We have created a corpus consisting of 712 German and 708 corresponding Simple German articles from seven web sources spanning different topics. Few Simple German articles are matched to multiple German ones, and vice versa. The parsed articles are structured by their respective source. Inside each source folder there is a json file with an entry per article containing all metadata consisting of the url, the crawling date, the publishing date (if available), a flag whether the article from this url is in simple language or not, and a list of all associated articles. Additionally, through the proposed automatic sentence alignment, we obtain a collection of about 5 942 instances in the form of sentence pairs, where one German sentence can have \( n \) Simple German sentences. We will talk about the quality of the sentence alignment in subsection 6.2. Inside the folder of sentence alignments there exist two files for each article. One file containing all aligned sentences in German and the other...
Table 1: Statistics of the article-aligned corpus. (a) articles per source, (t) tokens per source; all
further values are reported as averages: (s) sentences per article, (wps) words per sentence, and (w)
number of different words per article.

<table>
<thead>
<tr>
<th></th>
<th>Simple German</th>
<th></th>
<th>German</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>a</td>
<td>t</td>
<td>s</td>
</tr>
<tr>
<td>apo</td>
<td>168</td>
<td>94808</td>
<td>77.7</td>
</tr>
<tr>
<td>beb</td>
<td>21</td>
<td>5490</td>
<td>31.0</td>
</tr>
<tr>
<td>bra</td>
<td>47</td>
<td>9634</td>
<td>28.2</td>
</tr>
<tr>
<td>lmt</td>
<td>45</td>
<td>6946</td>
<td>20.0</td>
</tr>
<tr>
<td>mdr</td>
<td>322</td>
<td>53277</td>
<td>21.3</td>
</tr>
<tr>
<td>soz</td>
<td>15</td>
<td>5122</td>
<td>43.8</td>
</tr>
<tr>
<td>koe</td>
<td>82</td>
<td>66892</td>
<td>103.4</td>
</tr>
<tr>
<td>taz</td>
<td>8</td>
<td>7924</td>
<td>70.3</td>
</tr>
<tr>
<td>total</td>
<td>708</td>
<td>250093</td>
<td>49.5</td>
</tr>
</tbody>
</table>

the Simple German sentences at the corresponding line inside the file. Further, there is a json file
recording the name of the original article and the similarity value according to the alignment method.

Table 1 shows general statistics of the crawled and parsed articles, e.g. showcasing average sentence
length and the number of sentences per article. In general, Simple German articles tend to be
significantly shorter in average number of words per article, while the number of sentences is higher
in Simple German than in German articles. This may be due to the fact that long sentences in German
are split into multiple shorter, simpler sentences in Simple German. This emphasizes our decision to
allow n : 1 matches between Simple German and German sentences.

3.2 Licensing and Access

We publish the dataset under the CC BY-SA 4.0 license as a url collection and the accompanying
code to easily recreate the dataset under MIT license. In order to ensure the long-term availability
of the sources, we archived them in the Internet Archive4. We further share the entire, ready-to-use
dataset upon request via email.

4 Dataset Construction

In the following section we describe the process of data acquisition from the selection of the online
sources over the crawling of the websites to the parsing of the text. To be transparent, we point out
the problems and pitfalls that we experienced during the process.

Crawling  Starting point for the construction of the dataset was a set of websites. Since there is
no complete list of German websites in simple language, we produced the websites described in
Table 3 by conducting our own research. These websites are publicly available, offer parallel articles
in German and Simple German, and cover a range of different topics. Many websites offer content in
simple language, but few offer the same content additionally in simple language. Hence, we ignored
websites with content in only simple language. Due to its prevalence, most of the articles in our
dataset are written in “Leichte Sprache”, but as previously explained, we also included one website in
“Einfache Sprache” to increase the overall vocabulary size. Overall, the data collection was limited by
the availability of suitable and accessible data.

First, we identified a starting point for each website that offered an overview of all Simple German
articles which provided access to the crawler. Then, a crawling template for each website was created
using the python library BeautifulSoup5. The crawler always started from the articles in Simple

4 https://archive.org/web/
5 https://www.crummy.com/software/BeautifulSoup/
German and had to be adapted for each website to find the link to the corresponding article in German. We first crawled the entire webpage and later on parsed the text from the raw HTML-file, allowing for changes to the parsing without additional traffic for the website in question. Additionally, this process allows to return to the raw data to support unanticipated future uses.

**Parsing** We have ignored images, HTML-tags, and corresponding text metadata (e.g. bold writing, paragraph borders) for each article. In contrast to Aumiller and Gertz [1], where enumerations are removed since they may only contain single words or grammatically incorrect sentences, we decided to transform them into comma-separated text. Enumerations are frequently used in Simple German articles and we argue that they may contain major parts of information.

The most common challenge during crawling was an inconsistency in text location within a website, i.e. the structure of the HTML-boxes containing the main content. Simply extracting by `<p>`-tag was not sufficient, as these regularly contained useless footer information. As only the main text was the targeted resource, the crawler’s implementation needed to be unspecific enough to account for these deviations, but specific enough not to crawl any redundant or irrelevant website texts.

Another problem was the way in which the German articles and their corresponding translations in Simple German were linked. The (mdr), a state-funded public news organization, often showed inconsistent linking between articles. Here one might expect a strict structure disallowing differences. However, the links were sometimes encapsulated within `<a href>`-, sometimes given as plain text or not at all. The referenced German article could even be a video, rendering both articles useless for our corpus. Considering the technical limitations of the scraper, we had to discard Simple German articles whenever the original German source was unusable, e.g. unlocatable or in video format.

The result of the data acquisition as described above is a dataset of articles in German with their corresponding articles in Simple German.

5 Sentence Alignment

In the following section we compare different similarity measures and matching algorithms used to reach sentence-level alignment. We describe an article $A$ as a list of sentences, i.e. $A = [s_1, \ldots, s_n]$. We define $A^S$ and $A^C$ as the simple and complex versions of the same article with $|A^S| = n$ and $|A^C| = m$. We consider a variant of the sentence alignment problem that receives two lists of sentences $A^S$ and $A^C$ and produces a list of pairs $[(s^S_i, s^C_j)]$ such that, with relative certainty, $s^S_i$ is a (partial) simple version of the complex sentence $s^C_j$. We will approach this task by splitting it into three parts:

In a first step (Sec. 5.1), we transform the raw texts obtained in Section 4 into lists of sentences and do some light pre-processing. Next, we compute sentence similarity scores (Sec. 5.2) for pairs of sentences from the aligned articles. Finally, a sentence matching algorithm (Sec. 5.3) takes the lists of sentences and the respective inter-sentences similarities to calculate the most probable alignment.

5.1 Text Pre-processing

We choose to apply a number of pre-processing steps in order to facilitate the sentence matching. We neither apply lemmatization to the words nor do we remove stop words. The sentence borders are identified using spaCy [9] tools such that dates or ordinal numbers (usually followed by a period) do not lead to a wrong split of sentences. Also, all punctuation, including hyphens between compound nouns in Simple German, is removed from the text.

**Lowercase** letters are used whenever we use similarity measures that are based on TF-IDF to decrease the vocabulary size. However, for similarity measures based on word vectors we apply no conversion, because the word vectors actually differ between lowercase and uppercase letters, e.g. “essen” (to eat) and “Essen” (food), and for some nouns, spaCy returns a zero embedding vector for their lowercase version, e.g. “Mahnmal” (memorial).
Gender-conscious suffixes are removed. We are referring to word endings used in inclusive language to address female as well as other genders, not to endings that transform male nouns into their female form. In German, the female version of a word is often formed by appending “-in” (singular) or “-innen” (plural) to the end of the word, e.g. “der Pilot” (the male pilot) and “die Pilotin” (the female pilot). Traditionally, when talking about a group of people of unspecified gender, the male version was used. However, in order to include both men and women as well as other gender identities different endings are preferred. The most popular ones are using an uppercase I (“PilotIn”), a colon (“Pilot:in”), an asterisk (“Pilot*in”) and an underscore (“Pilot_in”). We remove these endings to make sentence matching easier. Such endings are commonly not included in Simple German texts.

5.2 Similarity Measures

After obtaining pre-processed lists of sentences $A^S$ and $A^C$, we compute similarities between any two sentences $s_i^S \in A^S$ and $s_j^C \in A^C$. A sentence can be described either as a list of words $s_i^S = [w_1^S, \ldots, w_l^S]$ or as a list of characters $s_i^C = [c_1^C, \ldots, c_k^C]$. In total, we have compared seven different similarity measures. Two of the measures are based on TF-IDF, the other five rely on word or sentence embeddings. We have decided to use the pre-trained word embeddings supplied by spaCy in the d_core_news_lg bundle and the pre-trained distiluse-base-multilingual-cased-v1 model provided by the authors⁷. We give a short overview of all similarity measures, refer to Appendix C for a detailed description.

TF-IDF based similarity measures  Both similarity measures calculate the cosine similarity $\cos_{\text{sim}}$ between two sentence vectors. We use the bag of word similarity [22] that represents each sentence as a bag of word vector weighted by calculating for each $w \in s_i$ the respective TF-IDF value. The character 4-gram similarity [32] works analogously, but instead of taking into account the words, it uses character n-grams. Following McNamee and Mayfield [18], we use $n = 4$.

Embedding based similarity measures  Using the pre-calculated word embeddings, the cosine similarity calculates the angle between the average of each sentence’s word vectors [32, 19]. The average similarity [12] calculates the average cosine similarity between all pairs of words in a given pair $(A^S, A^C)$ using the embedding vector $\text{emb}(w)$ of each word $w$. In contrast, the Continuous Word Alignment-based Similarity Analysis (CWASA) [8, 32] method does not average the embedding vectors. Instead, it finds the best matches for each word in $s^S$ and in $s^C$ with $\cos_{\text{sim}} \geq 0$. Then, the average cosine similarity is calculated between only the best matches. Likewise, the maximum similarity [12] calculates best matches for the words in both sentences. In contrast to CWASA, only the maximum similarity for each word in a sentence is considered. Further, we implement the bipartite similarity [12] that calculates a maximum matching on the weighted bipartite graph induced by the lists of simple and complex words. Edges between word pairs are weighted with the word-to-word cosine similarity. The method returns the average value of the edge weights in the maximum matching. The size of the maximum matching is bounded by the size of the smaller sentence. Finally, we implement the SBERT similarity by using a pre-trained multilingual SBERT model [23, 37] to create contextualized sentence embeddings and calculate the cosine similarity between them.

5.3 Matching Algorithms

The previously presented methods can be used to compute sentence similarity values for pairs of sentences. The sentence matching algorithm takes these values and determines which sentences are actual matches, i.e. translations. For two articles $A^S$ and $A^C$, $|A^S| = n$, $|A^C| = m$, the matrix $M \in \mathbb{R}^{n \times m}$ contains the sentence similarity measure for the sentences $s_i^S$ and $s_j^C$ in entry $M_{ij}$.

The goal is an $n : 1$ matching of various Simple German sentences to one German sentence, but not the other way round. We explain the reasoning for this in Section 3.1.

⁷https://spacy.io/models/de#de_core_news_lg
⁸https://www.sbert.net/
We compare two matching methods presented by Štajner et al. [32]. The first one is the most similar text algorithm (MST) which takes \( M \) and matches each \( s_i^S \in A^S \) with its closest sentence in \( A^C \).

The second method is the MST with Longest Increasing Sequence (MST-LIS). It is based on the observation that in many cases, the order of information stays the same in the simple and original article. It first uses MST. From this, only those matches appearing successively in the longest sequence are kept. All other simple sentences not contained in that sequence are included in a set of unmatched sentences. Let \((s_i^S, s_k^C), (s_j^S, s_l^C)\) be two matches in the longest sequence and \(i < j \Rightarrow k \leq l\). Then, for all unmatched sentences \( s_m^S \) with \(i < m < j\), a matching \( s_m^C \) will be looked for between indices \(k\) and \(l\). This is done iteratively, and for the potentially many sentences between \(s_i^S\) and \(s_j^S\) the corresponding matches cannot violate the original order in the Simple German article.

As an addition to the work of Štajner et al. [32], we introduce a threshold that defines a minimum similarity value for which two sentences should be matched. This leads to the desired result that simple sentences that do not have any corresponding complex sentences will likely not be matched at all, as they are expected to have a similarity lower than the threshold to all other sentences. An initial idea would be to pick a fixed value threshold as in Paetzold, Alva-Manchego, and Specia [22]. However, every sentence similarity method needs its own threshold, as their values are not in the same range, so that we decide upon a variable threshold. Let \(\mu\) be the mean of all sentence pair similarities, and let \(\sigma\) be their standard deviation. The threshold is then set to \(\mu(M) + k \cdot \sigma(M)\).

6 Evaluation

We combine both matching algorithms with all eight similarity measures using either a threshold of \(\mu + 1.5 \cdot \sigma\) or no threshold. This gives a total of 32 different alignment variants, the results of which we will discuss here. We select the best algorithm variant according to a two stage process. First, we analyse the results of the different alignment variants quantitatively. Then, we perform two kinds of manual evaluation. For the first one we create a ground truth by manually aligning the sentences for a subset of articles. The second one focuses on the matches by manually labelling them as either correct or incorrect alignments.

6.1 Quantitative Evaluation

In Table 2 we present – for all algorithm variants – the overall number of identified sentence matches, as well as the respective average similarity. Depending on the choice of similarity measure, matching method, and threshold, between 7 700 and 32 500 matched sentences pairs are found in the entire corpus of a total of 32 899 Simple German sentences. Introducing the threshold roughly halves the number of matches for the MST algorithm and results in only a third of matches for the MST-LIS...
Table 2: Match statistics for the full combination of settings.

<table>
<thead>
<tr>
<th>Alignment</th>
<th>Average Similarity</th>
<th>Sentence Matches</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MST</td>
<td>MST-LIS</td>
</tr>
<tr>
<td></td>
<td>- 1.5</td>
<td>- 1.5</td>
</tr>
<tr>
<td>bag of words</td>
<td>0.37</td>
<td>0.59</td>
</tr>
<tr>
<td>4-gram</td>
<td>0.45</td>
<td>0.68</td>
</tr>
<tr>
<td>cosine</td>
<td>0.68</td>
<td>0.77</td>
</tr>
<tr>
<td>average</td>
<td>0.17</td>
<td>0.21</td>
</tr>
<tr>
<td>CWASA</td>
<td>0.49</td>
<td>0.56</td>
</tr>
<tr>
<td>bipartite</td>
<td>0.65</td>
<td>0.71</td>
</tr>
<tr>
<td>maximum</td>
<td>0.56</td>
<td>0.62</td>
</tr>
<tr>
<td>sbert</td>
<td>0.49</td>
<td>0.57</td>
</tr>
</tbody>
</table>

Figure 3: (Left) Precision, recall and F1-score for all algorithm variants evaluated on the ground truth. (Right) Alignment classification accuracy for the manually labelled matches.

algorithm if the similarity measure is kept fixed. Using MST yields more matches than using MST-LIS, which is expected as the latter is more restrictive. Quite surprisingly, the average similarity of the matches is only a little lower for MST than for the MST-LIS for any fixed choice of similarity measure and threshold value. Consequently, the average similarity allows no conclusions about the quality of the matches. Further, we notice that using the similarity threshold always results in a higher average similarity.

Figure 2 gives an overview of the distributions of the similarity values over 100,000 randomly sampled sentence pairs for all similarity measures. In contrast to the TD-IDF based methods, all word embedding based methods show a different distribution. The average similarity measure is unimodally distributed, the other similarity measures show one distinct peak and small peak close to zero (cf. Figure 2b). However, the range of values and therefore the standard deviation seems to be especially small for average distance. The majority of the similarity values for the TF-IDF based methods (Figure 2a) is zero and we plot the corresponding graph with log-scale. This observation is intuitive, as the value of these sentence similarity strategies is always zero if the two evaluated sentences do not have a word (or 4-gram) in common.

6.2 Manual Evaluation

For our first evaluation we created a ground truth of sentence alignments by manually labelling a subset of articles, sampling uniformly 39 articles from the corpus. This allows us to evaluate the alignment algorithms with respect to precision, recall, and F1-score. To this end, we created a simple
GUI that presents the sentences of both articles side by side allowing us to find the n:1 matches of Simple German and German sentences. We consider additional simple sentences explaining difficult concepts as part of the alignment as long as they are a maximum of two sentences away from the literal translation of the German source sentence. We made the observation that depending on the source, the articles in Simple German are barely a translation of the original article. Also the order of information is often not maintained and in general, we only matched on average 33% of all German sentences. Figure 3 shows the results for all 32 algorithm variants on the ground truth. SBERT, bipartite and maximum similarity show good results. SBERT achieves the highest F1 score of 0.32 with precision and recall at 0.43 and 0.26, respectively. While maximum similarity achieves a lower F1 score with 0.45, the precision is higher.

Complementary to the first analysis, we continue by focusing only on the matches of each alignment algorithm. For the manual evaluation of the alignment we randomly sample 4,627 sentence pairs from the set of aligned sentences obtained from all algorithm variants. Given two sentences, it is inherently easier for a human annotator to make a yes/no decision whether the two presented sentences are a (partial) match or not. While this kind of evaluation does not allow any conclusions about the number of missed matches (i.e. recall) or the relation to additional explanatory sentences, we argue that it gives a different perspective on the quality of the computed alignments as done by [35]. As this analysis differs from the previous ground-truth set based analysis, we deliberately avoid the term precision and call the fraction of pairs that are labelled as (partial) matches as “manual alignment classification accuracy”. Thus, we create a different GUI only showing two sentences and asking the annotator to label them as either “match” (likewise for partial matches) or “no match”. The algorithm variant stays unknown to the user at evaluation time. Figure 3 (right) shows the results of the manual alignment classification accuracy analysis. The ranks of the algorithm variants roughly correspond to the ranks of under F1-score on the ground truth. Again, bipartite similarity, maximum similarity, and CWASA perform best. Maximum similarity with MST-LIS reaches the best manual alignment classification accuracy of 55.94%. Appendix D presents detailed results and a per website analysis.

Finally, we decide to create the sentence-level alignment using the maximum similarity metric with the MST-LIS matching, because it yields the highest precision on the ground truth and the highest manual alignment classification accuracy. Figure 1 shows exemplary alignments.

7 Discussion

The results for the sentence alignments presented in Section 5 show that the more sophisticated similarity measures perform better in terms of both F1-score and alignment classification accuracy. While the SBERT similarity is the most sophisticated similarity measure and yields the highest F1 score, it falls behind the precision and alignment classification accuracy of the maximum similarity with MST-LIS. Generally, MST-LIS benefits from its strong assumption about the order of information in both articles yielding a higher accuracy, but in return not finding all possible alignments. This can be traced back to our observation, that Simple German articles are often differently structured.

Limitations Our work presents a new dataset based on text data scraped from the internet. Hence, the quality of the text depends on the quality of the available online sources. Most of our data stems from the three sources apo, koe and mdr and thus providing a diverse vocabulary covered by our corpus. While this vocabulary covers a variety of different topics, we cannot rule out any negative side effects of data imbalance. Our dataset can only represent certain topics that were considered important enough to be translated into Simple German by the website in question.

In Subsection 6.2 we presented the different GUIs that we used to either manually align the sentence pairs or evaluate a sample of sentence alignments. One drawback of the tool for the second evaluation method is that it focuses solely on the matched sentences and presents them isolated from their contexts. One can argue that evaluators using the tool would have to see the context in which the sentences appear in order to correctly classify partial matches. Also, providing more information to the annotators might enable them to also correctly classify additional explanatory sentences.
Future Work The majority of our corpus encompasses data in “Leichte Sprache” and only one – though extensive – source in “Einfache Sprache”. A higher granularity of language difficulty could be achieved by incorporating texts originally directed at language learners that are rated, e.g. according to the European Reference System [7]. In any case, our work presents a parallel corpus for German and Simple German and should be continuously expanded. Not only to increase its size, but especially to increase the number topics covered in the corpus. However, as there do not seem to be any efforts to start a single big corpus like a Simple German Wikipedia, for the near future web scraping from various sources seems to be the method of choice. An additional option is to compute sentence alignments for existing article aligned corpora [e.g. 3] and to include them in the dataset.

As for the sentence alignment algorithms, various improvements are imaginable. Firstly, it might be interesting to allow one Simple German sentence to be matched to various German sentences. Also, for the matching algorithms we evaluated the effect of a threshold with \( k = 1.5 \). Further studies should examine the effects of different values for \( k \). The assumption of the MST-LIS about the order of information is very strong and a better recall could be achieved by softening the assumption and, e.g. allowing matches that are at most \( n \) sentences away. Other alignment algorithms that impose different biases on sentence order [2, 11, 38] are interesting for further extensions.

Our dataset can be used to train (or fine tune) automatic text simplification systems [e.g. 36] which then should produce text with properties similar to “Einfache Sprache”. Direct use cases for such simplification systems are support systems for human translators or browser plugins to simplify web pages [8]. Further research has shown that text simplification as a preprocessing step may increase performance in downstream natural language processing tasks such as information extraction [20], relation extraction [34], or machine translation [31]. It remains an interesting direction for future research if “Einfache Sprache” can help to further increase performance on such tasks.

8 Conclusion

In this paper, we presented a new monolingual sentence-aligned extendable corpus for Simple German – German that we make readily available. The data has been composed from eight different web sources and contains 708 aligned documents and a total of 5,942 aligned sentences. We have compared various similarity metrics and alignment methods from the literature and have introduced a variable similarity threshold that improves the sentence alignment.

We make the data accessible by releasing a URL collection [9] as well as the accompanying code for creating the dataset, i.e. code for text processing and sentence alignment. Our code can easily be adapted to create and analyze new sources. Even the application to non-German monolingual texts should be possible when specifying new word embeddings and adjusting the pre-processing steps.

We have obtained generally good results on our data. Our corpus is substantially bigger than the one in Klaper, Ebling, and Volk [13] (708 compared to 256 parallel articles) and our results of the best sentence alignment methods better as well (F1-scores: 0.28 compared to 0.085). It is also bigger than the parallel corpus created in Battisti et al. [3] (378 aligned documents), which does not provide any sentence level alignment.

Acknowledgments and Disclosure of Funding

This research has been funded by the Federal Ministry of Education and Research of Germany as part of the competence center for machine learning ML2R (01IS18038C).

8 See e.g. https://browser.mt/

9 To preserve the dataset for as long as possible, we have archived all articles using the WayBackMachine by the internet archive.
References


Checklist

1. For all authors...
   (a) Did the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A] Our work presents how a particular sentence-aligned dataset was created. It does not contain any theoretical claims.
   (b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments (e.g., for benchmarks)...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [No] Further explained in the Datasheet
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] Likewise discussed in Datasheet

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [No]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [No] All work for this article was done by persons that are listed among the authors of this paper. Part of this work has been done as a study project for which the students were given ECTS credit.