Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 1651–1664
Marseille, 20-25 June 2022

MUSS: Multilingual Unsupervised Sentence Simplification by Mining Paraphrases

Louis Martin[1,2] **Angela Fan**[1,3]
Éric de la Clergerie[2] **Antoine Bordes**[1] **Benoît Sagot**[2]
[1]Meta AI Research (Paris), [2]Inria (Paris), [3]LORIA (Nancy)
{louismartin, angelafan, abordes}@fb.com
{eric.de_la_clergerie, benoit.sagot}@inria.fr

Abstract

Progress in sentence simplification has been hindered by a lack of labeled parallel simplification data, particularly in languages other than English. We introduce MUSS, a Multilingual Unsupervised Sentence Simplification system that does not require labeled simplification data. MUSS uses a novel approach to sentence simplification that trains strong models using sentence-level paraphrase data instead of proper simplification data. These models leverage unsupervised pretraining and controllable generation mechanisms to flexibly adjust attributes such as length and lexical complexity at inference time. We show that this paraphrase data can be mined in any language from Common Crawl using semantic sentence embeddings, thus removing the need for labeled data. We evaluate our approach on English, French, and Spanish simplification benchmarks and closely match or outperform the previous best supervised results, despite not using any labeled simplification data. We push the state of the art further by incorporating labeled simplification data.

1. Introduction

Sentence simplification is the task of making a sentence easier to read and understand by reducing its lexical and syntactic complexity, while retaining most of its original meaning. Simplification has a variety of important societal applications, for example increasing accessibility for those with cognitive disabilities such as aphasia (Carroll et al., 1998), dyslexia (Rello et al., 2013), and autism (Evans et al., 2014), or for non-native speakers (Paetzold and Specia, 2016). Research has mostly focused on English simplification, where source texts and associated simplified texts exist and can be automatically aligned, such as English Wikipedia and Simple English Wikipedia (Zhang and Lapata, 2017). However, such data is limited in terms of size and domain, and difficult to find in other languages. Additionally, simplifying a sentence can be achieved in multiple ways, and depend on the target audience. Simplification guidelines are not uniquely defined, outlined by the stark differences in English simplification benchmarks (Alva-Manchego et al., 2020a). This highlights the need for more general models that can adjust to different simplification scenarios.

In this paper, we propose to train controllable models using sentence-level paraphrase data only, i.e. parallel sentences that have the same meaning but phrased differently. In order to generate simplifications and not paraphrases at test time, we use ACCESS (Martin et al., 2020) to control attributes such as length, lexical and syntactic complexity. Paraphrase data is more readily available, and opens the door to training flexible models that can adjust to more varied simplification scenarios. Our original goal was to mine simplifications from the web, but we surprisingly discovered that mining paraphrases leads to controllable models with better simplification performance while being more straightforward and requiring less prior assumptions (cf. Section 5.5). We gather such paraphrase data in any language by mining sentences from Common Crawl using semantic sentence embeddings. Simplification models trained on mined paraphrase data prove to work as well as models trained on large existing English paraphrase corpora (cf. Appendix D).

Our resulting Multilingual Unsupervised Sentence Simplification method, MUSS, is *unsupervised* because it can be trained without relying on *labeled* simplification data,[1] even though we mine using supervised sentence embeddings.[2] We apply MUSS on English, French, and Spanish to closely match or outperform the supervised state of the art in all languages. MUSS further improves the state of the art on all English datasets by incorporating additional labeled simplification data. We make the following contributions:

- We introduce a novel approach to training simplification models with paraphrase data only and propose a mining procedure to create large paraphrase corpora for any language.
- Our approach obtains strong performance. Without any labeled simplification data, we match or outperform the supervised state of the art in English, French and Spanish. We further improve the

[1] We use the term *labeled simplifications* to refer to parallel datasets where texts were manually simplified by humans.

[2] Previous works have also used the term *unsupervised simplification* to describe works that do not use any labeled parallel simplification data while leveraging supervised components such as constituency parsers and knowledge bases (Kumar et al., 2020), external synonymy lexicons (Surya et al., 2019), and databases of simplified synonyms (Zhao et al., 2020). We shall come back to these works in Section 2.

English state of the art by incorporating labeled simplification data.

- We release pretrained models, paraphrase data, and code for mining and training.[3]

	Type	# Sequence Pairs	# Avg. Tokens per Sequence
WikiLarge (English)	Labeled Parallel Simplifications	296,402	original: 21.7 simple: 16.0
Newsela (English)	Labeled Parallel Simplifications	94,206	original: 23.4 simple: 14.2
English	Mined	1,194,945	22.3
French	Mined	1,360,422	18.7
Spanish	Mined	996,609	22.8

Table 1: Statistics on our mined paraphrase training corpora compared to standard simplification datasets (see section 4.3 for more details).

2. Related work

Data-driven methods have been predominant in **English sentence simplification** in recent years (Alva-Manchego et al., 2020b), requiring large supervised training corpora of complex-simple aligned sentences (Wubben et al., 2012; Xu et al., 2016; Zhang and Lapata, 2017; Zhao et al., 2018; Martin et al., 2020). Methods have automatically aligned English and Simple English Wikipedia articles (Zhu et al., 2010; Coster and Kauchak, 2011; Woodsend and Lapata, 2011; Kauchak, 2013; Zhang and Lapata, 2017). Professional quality datasets such as Newsela (Xu et al., 2015) exist, but they are rare and come with restrictive licenses that hinder reproducibility and widespread usage.

Simplification in other languages has been explored in Brazilian Portuguese (Aluísio et al., 2008), Spanish (Saggion et al., 2015; Štajner et al., 2015), Italian (Brunato et al., 2015; Tonelli et al., 2016), Japanese (Goto et al., 2015; Kajiwara and Komachi, 2018; Katsuta and Yamamoto, 2019), and French (Gala et al., 2020). Research is hindered by a lack of large parallel corpora. In this work, we show that training on mined data can reach state-of-the-art results in each language. When labeled parallel simplification data is unavailable, systems rely on **unsupervised simplification**, often inspired from machine translation. The prevailing approach splits a monolingual corpora into sets of complex and simple sentences using readability metrics. Then simplification models are trained using automatic sentence alignments (Kajiwara and Komachi, 2016; Kajiwara and Komachi, 2018), auto-encoders (Surya et al., 2019; Zhao et al., 2020), unsupervised machine translation (Katsuta and Yamamoto, 2019), or back-translation (Aprosio et al., 2019). Other unsupervised approaches iteratively edit the sentence until a certain criterion is reached (Kumar et al., 2020), or use machine translation data to adapt English simplification models for other languages (Mallinson et al., 2020). The performance of unsupervised methods are often below their supervised counterparts. MUSS bridges the gap with supervised method and removes the need for deciding in advance how complex and simple sentences should be separated, but instead trains directly on paraphrases mined from the raw corpora.

Previous work on **parallel dataset mining** have been used mostly in machine translation using document retrieval (Munteanu and Marcu, 2005), language models (Koehn et al., 2018; Koehn et al., 2019), and embedding space alignment (Artetxe and Schwenk, 2019b) to create large corpora (Tiedemann, 2012; Schwenk et al., 2019). We focus on paraphrasing for sentence simplifications, which presents new challenges. Unlike machine translation, where the same sentence should be identified in two languages, we develop a method to identify varied paraphrases of sentences, that have a wider array of surface forms, including different lengths, multiple sentences, different vocabulary usage, and removal of content from more complex sentences.

Previous **unsupervised paraphrasing** research has aligned sentences from parallel corpora (Barzilay and Lee, 2003) with multiple objective functions (Liu et al., 2019). Bilingual pivoting relied on MT datasets to create large databases of word-level paraphrases (Pavlick et al., 2015), lexical simplifications (Pavlick and Callison-Burch, 2016; Kriz et al., 2018), or sentence-level paraphrase corpora (Wieting and Gimpel, 2018). This has not been applied to multiple languages or to sentence-level simplification. We also use raw monolingual data to create our paraphrase corpora instead of relying on parallel MT datasets.

3. Method

We now describe MUSS, our approach to training controllable simplification models on mined data.

3.1. Mining Paraphrases in Many Languages

Extracting Sequences Simplification consists of multiple rewriting operations, some of which span multiple sentences (e.g. sentence splitting or fusion). To allow such operations to be represented in our mined data, we extract chunks of text composed of multiple sentences that we call *sequences*.

We extract such sequences by first tokenizing a document into individual sentences $\{s_1, s_2, \ldots, s_n\}$ using NLTK (Bird and Loper, 2004). We then extract sequences of adjacent sentences with maximum length of 300 characters: $\{[s_1], [s_1, s_2], [s_1, \ldots, s_k], [s_2], [s_2, s_3], ...\}$. Noisy sequences are filtered out when they have more than 10% punctuation characters and when they have low language model probability according to a 3-gram language model trained with `kenlm` (Heafield, 2011) on Wikipedia.

Source texts are taken from CCNet (Wenzek et al., 2019), an extraction of Common Crawl (snapshot of

[3] https://github.com/facebookresearch/muss

the web). For English and French, we extract 1 billion sequences. For Spanish we extract 650 millions sequences, the maximum for this language in CCNet after filtering out noisy text.

Creating a Sequence Index Using Embeddings To automatically mine our paraphrase corpora, we first compute n-dimensional embeddings for each extracted sequence using LASER (Artetxe and Schwenk, 2019b). LASER provides joint multilingual sentence embeddings in 93 languages that have been successfully applied to the task of bilingual bitext mining (Schwenk et al., 2019). In this work, we show that LASER can also be used to mine monolingual paraphrase datasets but also highlights its limits (cf. Section 5.4). For each language we then index embeddings for each sequence using `faiss` (Johnson et al., 2019) for fast nearest neighbor search.

Mining Paraphrases We use each sequence as a query q_i against the billion-scale `faiss` index to retrieve the top-8 nearest neighbor in the LASER embedding space (L2 distance). We then use an upper bound on L2 distance and a margin criterion following (Artetxe and Schwenk, 2019a) to filter out nearest neighbors with low similarity. We refer the reader to Appendix Section A.1 for more technical details. The remaining nearest neighbors constitute a set of candidate aligned paraphrases to the query sequence: $\{(q_i, c_{i,1}), \ldots, (q_i, c_{i,k})\}$. We finally filter out poor alignments: sequences that are almost identical with Levenshtein distance $\leq 20\%$, sequences contained in one another, or that were extracted from two overlapping sliding windows of the same original document.

We report statistics of the mined corpora in English, French and Spanish in Table 1, examples of mined paraphrases in Appendix Table 7, and limits of this mining method in Section 5.4. Models trained on these mined paraphrases obtain similar performance than models trained on existing paraphrase datasets (cf. Appendix Section D).

3.2. Simplifying with ACCESS

In this section we describe how we adapt ACCESS (Martin et al., 2020) to train controllable sequence-to-sequence models on mined paraphrases, instead of labeled parallel simplifications. ACCESS is a method to make any sequence-to-sequence model controllable by conditioning on simplification-specific control tokens.

Training with Control Tokens At training time, the model is provided with control tokens that give oracle information on the target sequence, such as the amount of compression between the target and the source (length control). For example, when the target sequence is 80% of the length of the source sequence, the control token <NumChars_80%> is provided. At inference time generation can be controlled by selecting a given target control value. We adapt the original Levenshtein similarity control to only consider replace operations but otherwise use the same controls as (Martin et al., 2020). The controls used are: character length ratio, *replace-only* Levenshtein similarity, aggregated word frequency ratio, and dependency tree depth ratio. We thus prepend to every source in the training set the following 4 control tokens with sample-specific values: <NumChars_XX%> <LevSim_YY%> <WordFreq_ZZ%> <DepTreeDepth_TT%>. We refer the reader to the original paper (Martin et al., 2020) and Appendix A.2 for details on ACCESS and how those control tokens are computed.

Selecting Control Values at Inference After training with oracle controls, we can adjust them at inference to obtain the desired type of simplifications, for instance using shorter sentences for people with cognitive disabilities, or using more frequent words for second language learners. It is important that supervised and unsupervised simplification systems can be adapted to different conditions: (Kumar et al., 2020) choose operation-specific weights of their unsupervised simplification model for each benchmark and (Surya et al., 2019) select different models using SARI (Xu et al., 2016) on each validation set. Similarly, we set the 4 control hyper-parameters of ACCESS using SARI on each validation set and keep them fixed for samples in the test set.[4]. These control hyper-parameters are intuitive and easy to interpret: when no validation set is available, they can also be set using prior knowledge on the task and still lead to solid performance (cf. Appendix C).

3.3. Leveraging Unsupervised Pretraining

We combine our controllable models with unsupervised pretraining. For English, we finetune the pretrained generative model BART (Lewis et al., 2019) with ACCESS control tokens on our newly created training corpora. BART is a pretrained sequence-to-sequence model that generalizes other recent pretrained methods such as BERT (Devlin et al., 2018) for encoder-decoder models. For non-English, we use its multilingual version MBART (Liu et al., 2020), pretrained on 25 languages.

4. Experimental Setting

We assess the performance of our approach on three languages: English, French, and Spanish. Technical details for mining and training can be found in Appendix Section A. In all our experiments, we report scores on the test sets averaged over 5 random seeds with 95% confidence intervals.

4.1. Baselines

In addition to comparisons with previous works, we implement multiple baselines to assess the performance of our models, especially for French and Spanish where no previous simplification systems have open-source implementations.

[4] Details in Appendix A.2

	ASSET (en)		TurkCorpus (en)		Newsela (en)	
	SARI ↑	FKGL ↓	SARI ↑	FKGL ↓	SARI ↑	FKGL ↓
Baselines and Gold Reference						
Gold Reference	44.87±0.36	6.49±0.15	40.04±0.30	8.77±0.08	—	—
Unsupervised Systems						
BTRLTS (Zhao et al., 2020)	33.95	7.59	33.09	8.39	37.22	3.80
UNTS (Surya et al., 2019)	35.19	7.60	36.29	7.60	—	—
RM+EX+LS+RO (Kumar et al., 2020)	36.67	7.33	37.27	7.33	**38.33**	2.98
MUSS (mined data only)	**42.65**±0.23	8.23±0.62	**40.85**±0.15	8.79±0.30	**38.09**±0.59	5.12±0.47
Supervised Systems						
EditNTS (Dong et al., 2019)	34.95	8.38	37.66	8.38	39.30	3.90
DMASS-DCSS (Zhao et al., 2018)	38.67	7.73	39.92	7.73	—	—
ACCESS(Martin et al., 2020)	40.13	7.29	41.38	7.29	—	—
MUSS (labeled data only)	**43.63**±0.71	6.25±0.42	**42.62**±0.27	6.98±0.95	**42.59**±1.00	2.74±0.98
MUSS (labeled + mined data)	**44.15**±0.56	6.05±0.51	**42.53**±0.36	7.60±1.06	**41.17**±0.95	2.70±1.00

Table 2: **Unsupervised and Supervised Sentence Simplification for English.** We display SARI and FKGL on ASSET, TurkCorpus and Newsela test sets for English. Supervised models are trained on WikiLarge for the first two test sets, and Newsela for the last. Best SARI scores within confidence intervals are in bold.

Identity The entire original sequence is kept unchanged and used as the simplification.

Truncation The original sequence is truncated to the first 80% words. It is a strong baseline according to standard simplification metrics.

Pivot We use machine translation to use English models in other languages. The source non-English sentence is translated to English, simplified with our best supervised English model, and translated back into the source language. For French and Spanish translation, we use CCMATRIX (Schwenk et al., 2019) to train Transformer models with LayerDrop (Fan et al., 2019). We use MUSS trained on mined data + WikiLarge as the English simplification model.

Gold Reference We report gold reference scores for multi-reference datasets ASSET and TurkCorpus. We evaluate each reference against all others in a leave-one-out scenario, and then average the scores.

4.2. Evaluation Metrics

We evaluate with the standard metrics SARI and FKGL (Kincaid et al., 1975). We report BLEU (Papineni et al., 2002) only in Appendix Table 11 due its dubious suitability for simplification (Sulem et al., 2018).

SARI Sentence simplification is commonly evaluated with SARI (Xu et al., 2016), which compares model-generated simplifications with the source sequence and gold references. It averages F1 scores for addition, keep, and deletion operations. We compute SARI with the EASSE simplification evaluation suite (Alva-Manchego et al., 2019).[5]

Baselines	ALECTOR (fr) SARI ↑	Newsela (es) SARI ↑
Identity	26.16	16.99
Truncate	33.44	27.34
Pivot	33.48±0.37	**36.19**±0.34
MUSS†	**41.73**±0.67	**35.67**±0.46

Table 3: **Unsupervised Sentence Simplification in French and Spanish.** We display SARI scores in French (ALECTOR) and Spanish (Newsela). Best SARI scores within confidence intervals are in bold. †MBART+ACCESS model.

FKGL We report readability scores using Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), that combines of sentence lengths and word lengths. FKGL was designed to be used on English texts only, we do not report it on French and Spanish.

4.3. Training Data

For all languages we use the mined data described in Table 1 as training data. In English we show that training with additional labeled simplification data leads to better performance. We use two labeled datasets: **WikiLarge** (Zhang and Lapata, 2017) and **Newsela** (Xu et al., 2015). WikiLarge is composed of 296k simplifications automatically aligned from English Wikipedia and Simple English Wikipedia. Newsela is a collection of news articles with professional simplifications, later aligned into 94k simplifications by (Zhang and Lapata,

[5] We use the EASSE library (Alva-Manchego et al., 2019) to compute SARI. We recompute scores using previous work's system predictions available in EASSE.

2017).[6]

4.4. Evaluation Data

English We evaluate our English models on **ASSET** (Alva-Manchego et al., 2020a), **TurkCorpus** (Xu et al., 2016) and **Newsela** (Xu et al., 2015). TurkCorpus and ASSET were created using the same 2000 valid and 359 test source sentences and they respectively contain 8 and 10 reference simplifications per source sentence. ASSET features more varied set of rewriting operations than TurkCorpus, and is considered simpler by human judges (Alva-Manchego et al., 2020a). For Newsela, we evaluate on the split from (Zhang and Lapata, 2017), which includes 1129 validation and 1077 test sentence pairs.

French We use the French **ALECTOR** dataset (Gala et al., 2020). ALECTOR is a collection of literary (tales, stories) and scientific (documentary) texts along with their manual document-level simplified versions. These documents were extracted from material available to French primary school pupils. We split the dataset in 450 validation and 416 test sentence pairs (see Appendix A.3 for details).

Spanish We use the **Spanish part of Newsela** (Xu et al., 2015). We use the alignments from (Aprosio et al., 2019), composed of 2794 validation and 2795 test sentence pairs. Even though sentences were aligned using the CATS simplification alignment tool (Štajner et al., 2018), some alignment errors remain and automatic scores should be taken with a pinch of salt.

5. Results

5.1. English Simplification

We report MUSS scores in Table 2. We also compare to other state-of-the-art supervised models: DMASS-DCSS (Zhao et al., 2018), EditNTS (Dong et al., 2019), ACCESS (Martin et al., 2020); and unsupervised models: UNTS (Surya et al., 2019), BTRLTS (Zhao et al., 2020), and RM+EX+LS+RO (Kumar et al., 2020).

MUSS Unsupervised Results On the ASSET benchmark, with no labeled simplification data, MUSS obtains a +5.98 SARI improvement with respect to previous unsupervised methods, and a +2.52 SARI improvement over the state-of-the-art supervised methods. For the TurkCorpus and Newsela datasets, the unsupervised MUSS approach achieves strong results, either outperforming or closely matching unsupervised and supervised previous works.

MUSS Supervised Results When incorporating labeled data from WikiLarge and Newsela, MUSS obtains state-of-the-art results on all datasets. Using labeled data along with mined data does not always help compared to training only with labeled data, especially with the Newsela training set. Newsela is a high quality dataset focused on the specific domain of news articles. It might not benefit from additional lesser quality mined data.

Examples of Simplifications Various examples from our unsupervised system are shown in Table 4. Examining the simplifications, we see reduced sentence length, sentence splitting, and simpler vocabulary usage. For example, the words *in the town's western outskirts* is changed into *near the town* and *aerial nests* is simplified into *nests in the air*. We also witnessed errors related factual consistency. For instance related to named entity hallucination or disappearance. Tackling this problem would be interesting for future work.

5.2. French and Spanish Simplification

Our unsupervised approach to simplification can be applied to any language. Similar to English, we create a corpus of paraphrases composed of 1.4 million sequence pairs in French and 1.0 million sequence pairs in Spanish (cf. Table 1). We replace the monolingual BART with its multilingual counterpart MBART, trained on 25 languages. We report the performance of models trained on the mined corpus in Table 3. Unlike English, where labeled parallel training data has been created using Simple English Wikipedia, no such datasets exist for French or Spanish. Similarly, no other simplification systems are available in these languages. We thus compare to the identity, truncation and pivot baselines.

Results MUSS outperforms our strongest baseline by +8.25 SARI for French, while matching the pivot baseline performance for Spanish.

Besides using state-of-the-art machine translation models, the pivot baseline relies on a strong backbone simplification model that has two advantages compared to the French and Spanish simplification model. First the simplification model of the pivot baseline was trained on labeled simplification data from WikiLarge, which obtains +1.5 SARI in English compared to training only on mined data. Second it uses the stronger monolingual BART model instead of MBART. In Appendix Table 11, we can see that MBART has a small loss in performance of 1.54 SARI compared to its monolingual counterpart BART, due to the fact that it handles 25 languages instead of one. Further improvements could be achieved by using monolingual BART models trained for French or Spanish, possibly outperforming the pivot baseline.

5.3. Human Evaluation

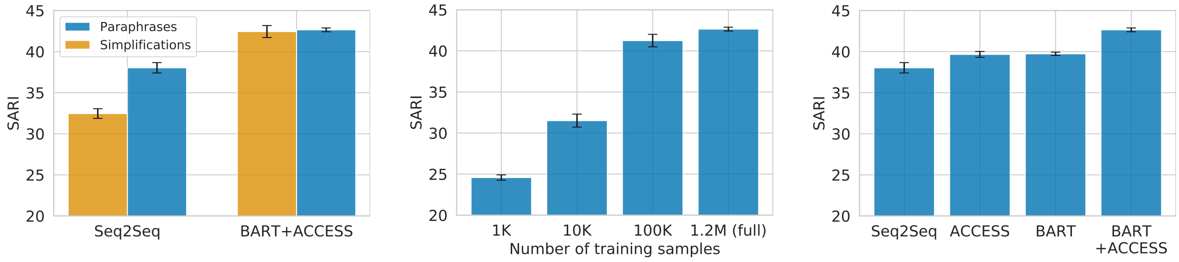
We further validate our approach with a human evaluation in all languages according to adequacy, fluency, and simplicity and report the results in Table 5.

Human Ratings Collection For human evaluation, we recruit volunteer native speakers for each language (5 in English, 2 in French, and 2 in Spanish). Simplification are evaluated on a 5 point Likert scale (0-4) ac-

[6] We experimented with other alignments (wiki-auto and newsela-auto (Jiang et al., 2020)) but with lower performance.

Original	**History** Landsberg prison, which is **in** the **town's western outskirts,** was **completed** in 1910.
Simplified	**The** Landsberg prison, which is **near** the **town,** was **built** in 1910.
Original	The name "hornet" is used for this and related species **primarily** because **of their habit of making aerial** nests **(similar to** the true hornets) rather than **subterranean nests.**
Simplified	The name "hornet" is used for this and related species because **they make** nests **in** the **air (like the** true hornets) rather than **in the ground.**
Original	Nocturnes is **an orchestral composition in three movements** by the French composer Claude Debussy.
Simplified	Nocturnes is **a piece of music for orchestra** by the French composer Claude Debussy.

Table 4: **Examples of Generated Simplifications.** We show simplifications generated by our best unsupervised model: MUSS trained on mined data only. Bold highlights differences between original and simplified.



(a) **Simplifications vs. Paraphrases**

(b) **Large-Scale Mining**

(c) **BART and ACCESS**

Figure 1: **Ablations** We display averaged SARI scores on the English ASSET test set with 95% confidence intervals (5 runs). (a) Models trained on mined simplifications or mined paraphrases, (b) MUSS trained on varying amounts of mined data, (c) Models trained with or without BART and/or ACCESS.

cording to: adequacy (*is the meaning preserved?*), fluency (*is the simplification fluent?*) and simplicity (*is the simplification actually simpler?*). For each system and each language, 50 simplifications are annotated. Each simplification is rated once. The simplifications are sampled from ASSET (English), ALECTOR (French), and Newsela (Spanish).

Discussion Table 5 displays the average ratings with 95% confidence intervals. Human judgments show that MUSS models are more fluent and produce simpler outputs than previous work (Martin et al., 2020). They are deemed as fluent and simpler than the human simplifications on ASSET test set, indicating that MUSS is able to reach a high level of simplicity thanks to the control mechanism. In French and Spanish, the unsupervised MUSS model performs better or similar than the supervised pivot baseline which has been trained on labeled English simplifications.

5.4. Fine-grained Analysis of MUSS Outputs

In table 6, we analyse the types of simplifications that MUSS performs using quality estimation features computed with the EASSE library. We decompose the SARI score into its three building blocks: F1 scores accounting for n-gram additions, deletions and keeps.

Copying the Source Over Simplification systems have suffered from not modifying the source sentence enough and often fall back to keeping it entirely unchanged (Wubben et al., 2012; Martin et al., 2020). MUSS on the other hand almost never resorts to exactly copying the source sentence which leads to higher addition and deletion F1.

Mined Data limits Sentence Splitting MUSS rarely perform sentence splitting when trained on mined data only (3.45% of the time) while it becomes way better at this operation when incorporating labelled data from WikiLarge (34.26%). Investigating the mined data reveals that our mining approach was not able to mined sentence splitting examples. Our intuition is that this is due to the fact that LASER embeddings do not work well across multiple sentences, thus preventing single sentences to be matched with multiple corresponding sentences. We identify mining sentence splitting examples as a promising direction of future work.

5.5. Ablations

Mining Simplifications vs. Paraphrases In this work, we mined paraphrases to train simplification models. This has the advantage of making fewer assumptions earlier on, by keeping the mining and models as general as possible.We also compared to directly mining simplifications using heuristics that enforces the target sentence to be simpler than the source similar to (Kajiwara and Komachi, 2016; Surya et al., 2019). We do so by first mining paraphrases without any constraints and then keeping only pairs that either contain sentence splits, reduced sequence length, or simpler vocabulary. We tuned these heuristics with validation SARI. The resulting dataset has 2.7 million simplification pairs. Figure 1a shows that seq2seq models trained on mined paraphrases achieve better performance. A similar trend exists with BART and ACCESS, con-

	English			**French**			**Spanish**		
	Adequacy	Fluency	Simplicity	Adequacy	Fluency	Simplicity	Adequacy	Fluency	Simplicity
ACCESS (Martin et al., 2020)	3.10±0.32	3.46±0.28	1.40±0.29	—	—	—	—	—	—
Pivot baseline	—	—	—	1.78±0.40	2.10±0.47	1.16±0.31	2.02±0.28	3.48±0.22	2.20±0.29
Gold Reference	**3.71**±0.18	3.78±0.18	1.78±0.30	**3.56**±0.21	**3.92**±0.10	**1.71**±0.32	**3.12**±0.29	**3.52**±0.25	1.70±0.46
MUSS (mined data)	3.20±0.28	3.84±0.14	1.88±0.33	2.88±0.34	3.50±0.32	1.22±0.25	2.26±0.29	3.48±0.25	**2.56**±0.29
MUSS (mined + labeled data)	3.12±0.34	**3.90**±0.14	**2.22**±0.36	—	—	—	—	—	—

Table 5: **Human Evaluation** Human ratings of adequacy, fluency and simplicity for ACCESS (Martin et al., 2020), pivot baseline, reference human simplifications, and MUSS. Scores are averaged over 50 ratings per system with 95% confidence intervals.

	Operation-specific SARI (F1 scores) ↑			**Quality Estimation** (%)		
	Additions	Deletions	Keeps	Exact Copies	Compression	Sent. Splits
BTRLTS (Zhao et al., 2020)	1.99	42.09	57.77	19.22	91.72	16.43
UNTS (Surya et al., 2019)	0.83	45.98	58.75	21.45	85.34	1.39
RM+EX+LS+RO (Kumar et al., 2020)	1.29	51.33	57.40	12.81	84.73	2.51
MUSS (mined data only)	8.09±0.74	60.87±0.61	59.00±0.48	0.11±0.19	88.61±7.16	3.45±2.31
EditNTS (Dong et al., 2019)	2.41	42.69	59.73	11.70	83.74	0.00
DMASS-DCSS (Zhao et al., 2018)	4.36	51.37	60.29	5.29	88.96	6.13
ACCESS(Martin et al., 2020)	6.54	50.85	62.99	4.18	94.08	20.89
MUSS (mined + labeled data)	11.14±0.34	60.40±1.64	60.90±1.30	0.11±0.19	88.92±3.34	34.26±12.97

Table 6: **Fine-grained Analysis of MUSS** We compare MUSS predictions with other systems on ASSET using the three operation-specific SARI components, % of simplifications which are exact copies of the source, average compression ratios, and % of simplifications with sentence splits.

firming that mining paraphrases can obtain better performance than mining simplifications.

How Much Mined Data Do You Need? In Figure 1b, we analyze the performance of training our best model on English on different amounts of mined data. By increasing the number of mined pairs, SARI drastically improves, indicating that efficient mining at scale is critical to performance. Unlike human-created training sets, unsupervised mining allows for large datasets in multiple languages.

Improvements from Pretraining and Control We compare the respective influence of pretraining BART and controllable generation ACCESS in Figure 1c. While both BART and ACCESS bring improvement over standard sequence-to-sequence, they work best in combination. Unlike previous approaches to text simplification, we use pretraining in our models. We find that the main qualitative improvement from pretraining is increased fluency and meaning preservation. For example, in Appendix Table 10, the model trained only with ACCESS substituted *culturally akin* with *culturally much like*, but when using BART, it is simplified to the more fluent *closely related*. While our mined data contains millions of sentences, pretraining methods are typically trained on billions thus enhancing the fluency and simplification performance.

6. Discussion

Since the pre-publication of this work, the pretrained model with controllable mechanism recipe has been explored in more details for English supervised simplification (Sheang and Saggion, 2021). Interestingly, the authors have found that using the T5 pretrained model (Raffel et al., 2020) in place of BART leads to an improvement in automatic metrics. Their T5+ACCESS setup trained on WikiLarge increases the performance with respectively 45.04 and 43.31 SARI on ASSET and TurkCorpus. They also highlight that adding control tokens to pretrained models leads to significant boosts in performance (up to +10.89 SARI) compared to using pretrained models only. Training controllable models in more settings and possibly other tasks thus appears as an interesting area of future work.

7. Conclusion

We propose a sentence simplification approach that does not rely on labeled parallel simplification data thanks to controllable generation, pretraining and large-scale mining of paraphrases from the web. This approach is language-agnostic and matches or outperforms previous state-of-the-art results, even from supervised systems that use labeled simplification data, on three languages: English, French, and Spanish. In future work, we plan to investigate how to scale this approach to more languages and types of simplification, and to apply this method to paraphrase generation. Another interesting direction for future work would to examine and improve factual consistency, especially related to named entity hallucination or disappearance.

Aluísio, S. M., Specia, L., Pardo, T. A., Maziero, E. G., and Fortes, R. P. (2008). Towards Brazilian Portuguese automatic text simplification systems. In *Proceedings of the eighth ACM symposium on Document engineering*, pages 240–248.

Alva-Manchego, F., Martin, L., Scarton, C., and Specia, L. (2019). EASSE: Easier automatic sentence simplification evaluation. In *EMNLP-IJCNLP: System Demonstrations*, November.

Alva-Manchego, F., Martin, L., Bordes, A., Scarton, C., Sagot, B., and Specia, L. (2020a). ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Procedings of ACL 2020*, pages 4668–4679.

Alva-Manchego, F., Scarton, C., and Specia, L. (2020b). Data-Driven Sentence Simplification: Survey and Benchmark. *Computational Linguistics*, pages 1–87.

Aprosio, A. P., Tonelli, S., Turchi, M., Negri, M., and Di Gangi, M. A. (2019). Neural text simplification in low-resource conditions using weak supervision. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 37–44.

Artetxe, M. and Schwenk, H. (2019a). Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy, July. Association for Computational Linguistics.

Artetxe, M. and Schwenk, H. (2019b). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Barzilay, R. and Lee, L. (2003). Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In *Proceedings of NAACL-HLT 2003*, pages 16–23.

Bird, S. and Loper, E. (2004). NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain.

Brunato, D., Dell'Orletta, F., Venturi, G., and Montemagni, S. (2015). Design and annotation of the first Italian corpus for text simplification. In *Proceedings of The 9th Linguistic Annotation Workshop*, pages 31–41.

Carroll, J., Minnen, G., Canning, Y., Devlin, S., and Tait, J. (1998). Practical simplification of English newspaper text to assist aphasic readers. In *Proceedings of the AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*, pages 7–10.

Coster, W. and Kauchak, D. (2011). Learning to simplify sentences using wikipedia. In *Proceedings of the workshop on monolingual text-to-text generation*, pages 1–9.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dong, Y., Li, Z., Rezagholizadeh, M., and Cheung, J. C. K. (2019). Editnts: An neural programmer-interpreter model for sentence simplification through explicit editing. In *Proceedings of ACL2019*, pages 3393–3402.

Evans, R., Orasan, C., and Dornescu, I. (2014). An evaluation of syntactic simplification rules for people with autism. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 131–140.

Fan, A., Grave, E., and Joulin, A. (2019). Reducing transformer depth on demand with structured dropout. *arXiv preprint arXiv:1909.11556*.

Gala, N., Tack, A., Javourey-Drevet, L., François, T., and Ziegler, J. C. (2020). Alector: A Parallel Corpus of Simplified French Texts with Alignments of Misreadings by Poor and Dyslexic Readers. In *Proceedings of LREC 2020*.

Goto, I., Tanaka, H., and Kumano, T. (2015). Japanese news simplification: Task design, data set construction, and analysis of simplified text. *Proceedings of MT Summit XV*.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.

Jiang, C., Maddela, M., Lan, W., Zhong, Y., and Xu, W. (2020). Neural crf model for sentence alignment in text simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Johnson, J., Douze, M., and Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Transactions Big Data*.

Kajiwara, T. and Komachi, M. (2016). Building a monolingual parallel corpus for text simplification using sentence similarity based on alignment between word embeddings. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1147–1158, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Kajiwara, T. and Komachi, M. (2018). Text simplification without simplified corpora. *The Journal of Natural Language Processing*, 25:223–249.

Katsuta, A. and Yamamoto, K. (2019). Improving text simplification by corpus expansion with unsupervised learning. In *2019 International Conference on Asian Language Processing (IALP)*, pages 216–221. IEEE.

Kauchak, D. (2013). Improving text simplification language modeling using unsimplified text data. In *Proceedings of ACL 2013*, pages 1537–1546.

Kincaid, J. P., Fishburne Jr., R. P., Rogers, R. L., and Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Koehn, P., Khayrallah, H., Heafield, K., and Forcada, M. L. (2018). Findings of the WMT 2018 shared task on parallel corpus filtering. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 726–739, October.

Koehn, P., Guzmán, F., Chaudhary, V., and Pino, J. (2019). Findings of the WMT 2019 shared task on parallel corpus filtering for low-resource conditions. In *Proceedings of the Fourth Conference on Machine Translation*, pages 54–72.

Kriz, R., Miltsakaki, E., Apidianaki, M., and Callison-Burch, C. (2018). Simplification using paraphrases and context-based lexical substitution. In *Proceedings of NAACL 2018*, pages 207–217. Association for Computational Linguistics, June.

Kumar, D., Mou, L., Golab, L., and Vechtomova, O. (2020). Iterative edit-based unsupervised sentence simplification. In *Proceedings of ACL 2020*, pages 7918–7928, Online, July.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Liu, X., Mou, L., Meng, F., Zhou, H., Zhou, J., and Song, S. (2019). Unsupervised paraphrasing by simulated annealing. *arXiv preprint arXiv:1909.03588*.

Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., Lewis, M., and Zettlemoyer, L. (2020). Multilingual denoising pre-training for neural machine translation. *arXiv preprint arXiv:2001.08210*.

Mallinson, J., Sennrich, R., and Lapata, M. (2020). Zero-shot crosslingual sentence simplification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5109–5126, Online, November. Association for Computational Linguistics.

Martin, L., Humeau, S., Mazaré, P.-E., de La Clergerie, É., Bordes, A., and Sagot, B. (2018). Reference-less quality estimation of text simplification systems. In *Proceedings of the 1st Workshop on Automatic Text Adaptation*, pages 29–38.

Martin, L., Sagot, B., de la Clergerie, É., and Bordes, A. (2020). Controllable sentence simplification. In *Proceedings of LREC 2020*.

Munteanu, D. S. and Marcu, D. (2005). Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504.

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. (2019). fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL 2019 (Demonstrations)*, pages 48–53, Minneapolis, Minnesota.

Paetzold, G. H. and Specia, L. (2016). Unsupervised lexical simplification for non-native speakers. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL 2002*, pages 311–318.

Pavlick, E. and Callison-Burch, C. (2016). Simple ppdb: A paraphrase database for simplification. In *Proceedings of ACL 2016*, pages 143–148.

Pavlick, E., Rastogi, P., Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2015). Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of ACL-CoLing 2015*.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Rapin, J. and Teytaud, O. (2018). Nevergrad - A gradient-free optimization platform. `https://GitHub.com/FacebookResearch/Nevergrad`.

Rello, L., Baeza-Yates, R., Bott, S., and Saggion, H. (2013). Simplify or help?: text simplification strategies for people with dyslexia. In *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility*, page 15. ACM.

Saggion, H., Štajner, S., Bott, S., Mille, S., Rello, L., and Drndarevic, B. (2015). Making it simplext: Implementation and evaluation of a text simplification system for spanish. *ACM Transactions on Accessible Computing*, 6(4):1–36.

Schwenk, H., Wenzek, G., Edunov, S., Grave, E., and Joulin, A. (2019). Ccmatrix: Mining billions of high-quality parallel sentences on the web. *arXiv:1911.04944*.

Sheang, K. C. and Saggion, H. (2021). Controllable sentence simplification with a unified text-to-text transfer transformer. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 341–352, Aberdeen, Scotland, UK, August. Association for Computational Linguistics.

Štajner, S., Calixto, I., and Saggion, H. (2015). Automatic text simplification for Spanish: Comparative evaluation of various simplification strategies. In *Proceedings of the international conference recent advances in natural language processing*.

Štajner, S., Franco-Salvador, M., Rosso, P., and Ponzetto, S. P. (2018). CATS: A tool for customized alignment of text simplification corpora. In

Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, May. European Language Resources Association (ELRA).

Sulem, E., Abend, O., and Rappoport, A. (2018). Semantic structural evaluation for text simplification. In *Proceedings of the NAACL-HLT 2018*, pages 685–696.

Surya, S., Mishra, A., Laha, A., Jain, P., and Sankaranarayanan, K. (2019). Unsupervised neural text simplification. In *Proceedings of ACL 2018*, pages 2058–2068, July.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of LREC 2012*.

Tonelli, S., Aprosio, A. P., and Saltori, F. (2016). SIMPITIKI: a Simplification corpus for Italian. *Proceedings of the Third Italian Conference on Computational Linguistics*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008.

Wenzek, G., Lachaux, M.-A., Conneau, A., Chaudhary, V., Guzman, F., Joulin, A., and Grave, E. (2019). CCnet: Extracting high quality monolingual datasets from web crawl data. *arXiv preprint arXiv:1911.00359*.

Wieting, J. and Gimpel, K. (2018). ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of ACL 2018*, pages 451–462, Melbourne, Australia.

Woodsend, K. and Lapata, M. (2011). Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of EMNLP 2011*, pages 409–420, Edinburgh, Scotland, UK.

Wubben, S., Van Den Bosch, A., and Krahmer, E. (2012). Sentence simplification by monolingual machine translation. In *Proceedings of ACL 2012*, pages 1015–1024.

Xu, W., Callison-Burch, C., and Napoles, C. (2015). Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, pages 283–297.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., and Callison-Burch, C. (2016). Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Zhang, X. and Lapata, M. (2017). Sentence simplification with deep reinforcement learning. In *Proceedings of EMNLP 2017*, pages 584–594, Copenhagen, Denmark.

Zhao, S., Meng, R., He, D., Saptono, A., and Parmanto, B. (2018). Integrating transformer and paraphrase rules for sentence simplification. In *Proceedings of EMNLP 2018*, pages 3164–3173, Brussels, Belgium, October-November.

Zhao, Y., Chen, L., Chen, Z., and Yu, K. (2020). Semi-supervised text simplification with back-translation and asymmetric denoising autoencoders. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Zhu, Z., Bernhard, D., and Gurevych, I. (2010). A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1353–1361.

Appendices

A. Experimental details

In this section we describe specific details of our experimental procedure. Figure 2 is a overall reminder of our method presented in the main paper.

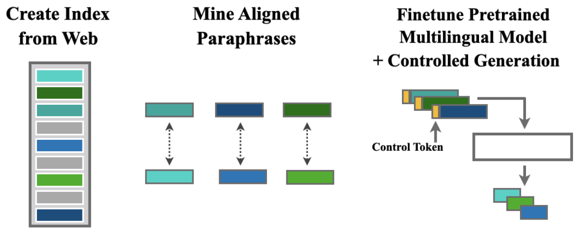

Create Index from Web
Mine Aligned Paraphrases
Finetune Pretrained Multilingual Model + Controlled Generation
Control Token

Figure 2: **Sentence Simplification Models for Any Language without Simplification Data**. Sentences from the web are used to create a large scale index that allows mining millions of paraphrases. Subsequently, we finetune pretrained models augmented with controllable mechanisms on the paraphrase corpora to achieve sentence simplification models in any language.

A.1. Mining Details

Sequence Extraction We only consider documents from the HEAD split in CCNet— this represents the third of the data with the best perplexity using a language model.

Paraphrase Mining We compute LASER embeddings of dimension 1024 and reduce dimensionality with a 512 PCA followed by random rotation. We further compress them using 8 bit scalar quantization. The compressed embeddings are then stored in a `faiss` inverted file index with 32,768 cells (nprobe=16). These embeddings are used to mine pairs of paraphrases. We return the top-8 nearest neighbors, and keep those with L2 distance lower than 0.05 and relative distance compared to other top-8 nearest neighbors lower than 0.6.

Paraphrases Filtering The resulting paraphrases are filtered to remove almost identical paraphrases by enforcing a case-insensitive character-level Levenshtein distance (Levenshtein, 1966) greater or equal to 20%. We remove paraphrases that come from the same document to avoid aligning sequences that overlapped each other in the text. We also remove paraphrases where one of the sequence is contained in the other. We further filter out any sequence that is present in our evaluation datasets.

A.2. Training Details

Seq2Seq training We implement our models with `fairseq` (Ott et al., 2019). All our models are Transformers (Vaswani et al., 2017) based on the BART$_{\text{Large}}$ architecture (388M parameters), keeping the optimization procedure and hyper-parameters fixed to those used in the original implementation (Lewis et al., 2019)[7]. We either randomly initialize weights for the standard sequence-to-sequence experiments or initialize with pretrained BART for the BART experiments. When initializing the weights randomly, we use a learning rate of 3.10^{-4} versus the original 3.10^{-5} when finetuning BART. For a given seed, the model is trained on 8 Nvidia V100 GPUs during approximately 10 hours.

Controllable Generation For controllable generation, we use the open-source ACCESS implementation (Martin et al., 2018). We use the same control parameters as the original paper, namely length, Levenshtein similarity, lexical complexity, and syntactic complexity.[8]

As mentioned in Section "Simplifying with ACCESS", we select the 4 ACCESS hyper-parameters using SARI on the validation set. We use zero-order optimization with the NEVERGRAD library (Rapin and Teytaud, 2018). We use the OnePlusOne optimizer with a budget of 64 evaluations (approximately 1 hour of optimization on a single GPU). The hyper-parameters are contained in the $[0.2, 1.5]$ interval.

The 4 hyper-parameter values are then kept fixed for all sentences in the associated test set.

Translation Model for Pivot Baseline For the pivot baseline we train models on `ccMatrix` (Schwenk et al., 2019). Our models use the Transformer architecture with 240 million parameters with LayerDrop (Fan et al., 2019). We train for 36 hours on 8 GPUs following the suggested parameters in (Ott et al., 2019).

Gold Reference Baseline To avoid creating a discrepancy in terms of number of references between the gold reference scores, where we leave one reference out, and when we evaluate the models with all references, we compensate by duplicating one of the other references at random so that the total number of references is unchanged.

A.3. Evaluation Details

SARI score computation We use the latest version of SARI implemented in `EASSE` (Alva-Manchego et al., 2019) which fixes bugs and inconsistencies from the traditional implementation of SARI. As a consequence, we also recompute scores from previous systems that we compare to. We do so by using the system predictions provided by the respective authors, and available in `EASSE`.

[7] All hyper-parameters and training commands for `fairseq` can be found here: `https://github.com/pytorch/fairseq/blob/master/examples/bart/README.summarization.md`

[8] We modify the Levenshtein similarity parameter to only consider replace operations, by assigning a 0 weight to insertions and deletions. This change helps decorrelate the Levenshtein similarity control token from the length control token and produced better results in preliminary experiments.

ALECTOR Sentence-level Alignment The ALECTOR corpus comes as source documents and their manual simplifications but not sentence-level alignment is provided. Luckily, most of these documents were simplified line by line, each line consisting of a few sentences. For each source document, we therefore align each line, provided it is not too long (less than 6 sentences), with the most appropriate line in the simplified document, using the LASER embedding space. The resulting alignments are split into validation and test by randomly sampling the documents for the validation (450 sentence pairs) and rest for test (416 sentence pairs).

A.4. Links to Datasets

The datasets we used are available at the following addresses:

- CCNet: `https://github.com/facebookresearch/cc_net`.
- WikiLarge: `https://github.com/XingxingZhang/dress`.
- ASSET: `https://github.com/facebookresearch/asset` or `https://github.com/feralvam/easse`.
- TurkCorpus: `https://github.com/cocoxu/simplification/` or `https://github.com/feralvam/easse`.
- Newsela: This dataset has to be requested at `https://newsela.com/data`.
- ALECTOR: This dataset has to be requested from the authors (Gala et al., 2020).

B. Characteristics of the mined data

We show in Figure 3 the distribution of different surface features of our mined data versus those of WikiLarge. Some examples of mined paraphrases are shown in Table 7.

C. Set ACCESS Control Parameters Without Parallel Data

In our experiments we adjusted our model to the different dataset conditions by selecting our ACCESS control tokens with SARI on each validation set. When no such parallel validation set exists, we show that strong performance can still be obtained by using prior knowledge for the given downstream application. This can be done by setting all 4 ACCESS control hyper-parameters to an intuitive guess of the desired compression ratio.

To illustrate this for the considered evaluation datasets, we first independently sample 50 source sentences and 50 random *unaligned* simple sentences from each validation set. These two groups of non-parallel sentences are used to approximate the character-level compression ratio between complex and simplified sentences. We do so by dividing the average length of the simplified sentences by the average length of the 50 source sentences. We finally use this approximated compression ratio as the value of all 4 ACCESS hyper-parameters. In practice, we obtain the following approximations: ASSET $= 0.8$, TurkCorpus $= 0.95$, and Newsela $= 0.4$ (rounded to 0.05). Results in Table 8 show that the resulting model performs very close to when we adjust the ACCESS hyper-parameters using SARI on the complete validation set.

D. Comparing to Existing Paraphrase Datasets

We compare using our mined paraphrase data with existing large-scale paraphrase datasets in Table 9. We use PARANMT (Wieting and Gimpel, 2018), a large paraphrase dataset created using back-translation on an existing labeled parallel machine translation dataset. We use the same 5 million top-scoring sentences that the authors used to train their sentence embeddings. Training MUSS on the mined data or on PARANMT obtains similar results for text simplification, confirming that mining paraphrase data is a viable alternative to using existing paraphrase datasets relying on labeled parallel machine translation corpora.

E. Influence of BART on Fluency

In Table 10, we present some selected samples that highlight the improved fluency of simplifications when using BART.

F. Additional Scores

BLEU We report additional BLEU scores for completeness. These results are displayed along with SARI and FKGL for English. These BLEU scores should be carefully interpreted. They have been found to correlate poorly with human judgments of simplicity (Sulem et al., 2018). Furthermore, the identity baseline achieves very high BLEU scores on some datasets (e.g. 92.81 on ASSET or 99.36 on TurkCorpus), which underlines the weaknesses of this metric.

Validation Scores We report English validation scores to foster reproducibility in Table 12.

Seq2Seq Models on Mined Data When training a Transformer sequence-to-sequence model (Seq2Seq) on WikiLarge compared to the mined corpus, models trained on the mined data perform better. It is surprising that a model trained solely on paraphrases achieves such good results on simplification benchmarks. Previous works have shown that simplification models suffer from not making enough modifications to the source

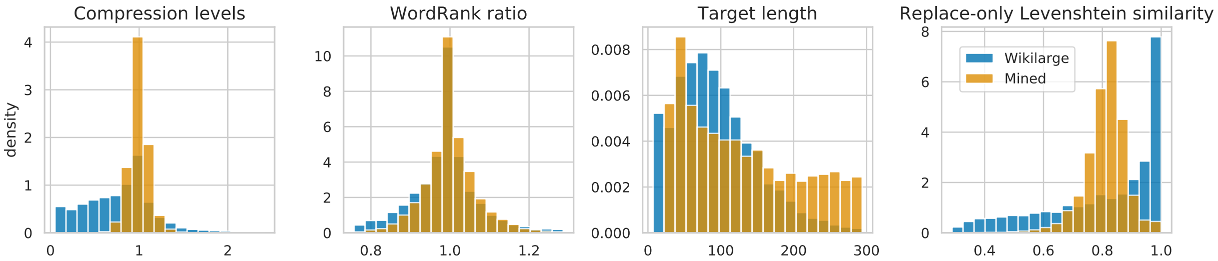


Figure 3: Density of several text features in WikiLarge and our mined data. The WordRank ratio is a measure of lexical complexity reduction (Martin et al., 2020). Replace-only Levenshtein similarity only considers replace operations in the traditional Levenshtein similarity and assigns 0 weights to insertions and deletions.

Query	**For insulation, it uses foam-injected** polyurethane which helps **ensure** the quality of the ice **produced by the machine. It comes with** an easy to clean air filter.
Mined	**It has** polyurethane **for insulation** which **is foam-injected. This** helps **to maintain** the quality of the ice **it produces. The unit has** an easy to clean air filter.
Query	Here are some **useful** tips and **tricks** to **identify and** manage your **stress.**
Mined	Here are some tips and **remedies you can follow** to manage **and control** your **anxiety.**
Query	**As cancer** cells **break apart,** their contents are **released** into the **blood.**
Mined	**When brain** cells **die,** their contents are **partially spilled back** into the **blood in the form of debris.**
Query	**The trail** is ideal for **taking** a short **hike with small children** or a **longer, more rugged overnight trip.**
Mined	**It** is **the** ideal **location** for a short **stroll, a nature walk** or a **longer walk.**
Query	Thank you for **joining us, and please** check **out** the **site.**
Mined	Thank you for **calling us. Please** check the **website.**

Table 7: **Examples of Mined Paraphrases**. Paraphrases, although sometimes not preserving the entire meaning, display various rewriting operations, such as lexical substitution, compression or sentence splitting.

	ASSET	**TurkCor.**	**Newsela**
Method	SARI ↑	SARI ↑	SARI ↑
SARI on valid	42.65±0.23	40.85±0.15	38.09±0.59
Approx. value	42.49±0.34	39.57±0.40	36.16±0.35

Table 8: **Set ACCESS Controls Wo. Parallel Data**
Setting ACCESS parameters of MUSS +MINED model either using SARI on the validation set or using only 50 *unaligned* sentence pairs from the validation set. All ACCESS parameters are set to the same approximated value: ASSET = 0.8, TurkCorpus = 0.95, and Newsela = 0.4).

	ASSET	**TurkCor.**	**Newsela**
Data	SARI ↑	SARI ↑	SARI ↑
MINED	42.65±0.23	40.85±0.15	38.09±0.59
PARANMT	42.50±0.33	40.50±0.16	39.11±0.88

Table 9: **Mined Data vs. ParaNMT**
We compare SARI scores of MUSS trained either on our mined data or on PARANMT (Wieting and Gimpel, 2018) on the test sets of ASSET, TurkCorpus and Newsela.

sentence and found that forcing models to rewrite the input was beneficial (Wubben et al., 2012; Martin et al., 2020). This is confirmed when investigating the F1 deletion component of SARI which is 20 points higher for the model trained on paraphrases.

Original	They are culturally akin to the coastal peoples of Papua New Guinea.
ACCESS	**They're** culturally **much like** the Papua New **Guinea coastal peoples.**
BART+ACCESS	They are **closely related** to coastal **people** of Papua New Guinea
Original	Orton and his wife welcomed Alanna Marie Orton on July 12, 2008.
ACCESS	Orton and his wife **had been called** Alanna Marie Orton on July **12**.
BART+ACCESS	Orton and his wife **gave birth to** Alanna Marie Orton on July 12, 2008.
Original	He settled in London, devoting himself chiefly to practical teaching.
ACCESS	He **set up** in **London and made** himself **mainly for** teaching.
BART+ACCESS	He settled in **London and devoted** himself to teaching.

Table 10: **Influence of BART on Simplifications.** We display some examples of generations that illustrate how BART improves the fluency and meaning preservation of generated simplifications.

	Data	**ASSET**			**TurkCorpus**			**Newsela**		
Baselines and Gold Reference		SARI ↑	BLEU ↑	FKGL ↓	SARI ↑	BLEU ↑	FKGL ↓	SARI ↑	BLEU ↑	FKGL ↓
Identity Baseline	—	20.73	92.81	10.02	26.29	99.36	10.02	—	—	—
Truncate Baseline	—	29.85	84.94	7.91	33.10	88.82	7.91	—	—	—
Reference	—	44.87±0.36	68.95±1.33	6.49±0.15	40.04±0.30	73.56±1.18	8.77±0.08	—	—	—
Supervised Systems (This Work)										
Seq2Seq	WikiLarge	32.71±1.55	88.56±1.06	8.62±0.34	35.79±0.89	90.24±2.52	8.63±0.34	22.23±1.99	21.75±0.45	8.00±0.26
MUSS	WikiLarge	43.63±0.71	76.28±4.30	6.25±0.42	42.62±0.27	78.28±3.95	6.98±0.95	40.00±0.63	14.42±6.85	3.51±0.53
MUSS	WikiLarge + Mined	44.15±0.56	72.98±4.27	6.05±0.51	42.53±0.36	78.17±2.20	7.60±1.06	39.50±0.42	15.52±0.99	3.19±0.49
MUSS	Newsela	42.91±0.58	71.40±6.38	6.91±0.42	41.53±0.36	74.29±4.67	7.39±0.42	42.59±1.00	18.61±4.49	2.74±0.98
MUSS	Newsela + Mined	41.36±0.48	78.35±2.83	6.96±0.26	40.01±0.51	83.77±1.00	8.26±0.36	41.17±0.95	16.87±4.55	2.70±1.00
Unsupervised Systems (This Work)										
Seq2Seq	Mined	38.03±0.63	61.76±2.19	9.41±0.07	38.06±0.47	63.70±2.43	9.43±0.07	30.36±0.71	12.98±0.32	8.85±0.13
MUSS (mBART)	Mined	41.11±0.70	77.22±2.12	7.18±0.21	39.40±0.54	77.05±3.02	8.65±0.40	34.76±0.96	19.06±1.15	5.44±0.25
MUSS (BART)	Mined	42.65±0.23	66.23±4.31	8.23±0.62	40.85±0.15	63.76±4.26	8.79±0.30	38.09±0.59	14.91±1.39	5.12±0.47

Table 11: **Detailed English Results.** We display SARI, BLEU, and FKGL on ASSET, TurkCorpus and Newsela English evaluation datasets (test sets).

	Data	**ASSET**			**TurkCorpus**			**Newsela**		
Baselines and Gold Reference		SARI ↑	BLEU ↑	FKGL ↓	SARI ↑	BLEU ↑	FKGL ↓	SARI ↑	BLEU ↑	FKGL ↓
Identity Baseline	—	22.53	94.44	9.49	26.96	99.27	9.49	12.00	20.69	8.77
Truncate Baseline	—	29.95	86.67	7.39	32.90	89.10	7.40	24.64	18.97	6.90
Reference	—	45.22±0.94	72.67±2.83	6.13±0.56	40.66±0.11	77.21±0.45	8.31±0.04	—	—	—
Supervised Systems (This Work)										
Seq2Seq	WikiLarge	33.87±1.90	90.21±1.14	8.31±0.34	35.87±1.09	91.06±2.24	8.31±0.34	20.89±4.08	20.97±0.53	8.27±0.46
MUSS	WikiLarge	45.58±0.28	78.85±4.44	5.61±0.31	43.26±0.42	78.39±3.08	6.73±0.38	39.66±1.80	14.82±7.17	4.64±1.85
MUSS	WikiLarge + Mined	45.50±0.69	73.16±4.41	5.83±0.51	43.17±0.19	77.52±3.01	7.19±1.02	40.50±0.56	16.30±0.97	3.57±0.60
MUSS	Newsela	43.91±0.10	70.06±10.05	6.47±0.29	41.94±0.21	74.03±7.51	6.99±0.53	42.36±1.32	19.18±6.03	3.20±1.01
MUSS	Newsela + Mined	42.48±0.41	77.86±3.13	6.41±0.13	40.77±0.52	83.04±1.16	7.68±0.30	41.68±1.60	17.23±5.28	2.97±0.91
Unsupervised Systems (This Work)										
Seq2Seq	Mined	38.88±0.22	61.80±0.94	8.63±0.13	37.51±0.10	62.04±0.91	8.64±0.13	30.35±0.23	13.04±0.45	8.87±0.12
MUSS (mBART)	Mined	41.68±0.72	77.11±2.02	6.56±0.21	39.60±0.44	75.64±2.85	8.04±0.40	34.59±0.59	18.19±1.26	5.76±0.22
MUSS (BART)	Mined	43.01±0.23	67.65±4.32	7.75±0.53	40.61±0.18	63.56±4.30	8.28±0.18	38.07±0.22	14.43±0.97	5.40±0.41

Table 12: **English Results on Validation Sets.** We display SARI, BLEU, and FKGL on ASSET, TurkCorpus and Newsela English datasets (validation sets).