The Role of Text Simplification Operations in Evaluation

Laura Vásquez-Rodríguez[1], Matthew Shardlow[2], Piotr Przybyła[3] and Sophia Ananiadou[1,4]

[1]National Centre for Text Mining, The University of Manchester, Manchester, United Kingdom
[2]Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, United Kingdom
[3]Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland
[4]The Alan Turing Institute, London, United Kingdom

Abstract

Research in Text Simplification (TS) has relied mostly on the Wikipedia-based datasets and the SARI evaluation metric, as the preferred means for creating and evaluating new simplification methods. Previous studies have pointed out the flaws of data evaluation resources, including incorrect alignment of simple/complex sentence pairs, sentences with no simplifications or a dearth in the variety of simplification operations. However, there are no further analyses on the impact of the original data distribution regarding the type of simplification operations performed. In this paper, we set up a systematic benchmark of the most common TS datasets, basing our evaluation on different protocols for split selection (e.g., selection by random or by Monte Carlo). We perform an operation-based investigation, demonstrating in detail the limitations of existing simplification datasets. Further, we make recommendations for future standardised practices in the design, creation and evaluation of TS resources.

Keywords

Text Simplification, Evaluation, Edit-operations, Simplification-operations, Wikipedia-based datasets

1. Introduction

TS methods transform complex text fragments into their simple variants, according to specific operations and audiences. Non-native speakers can significantly benefit from the substitution of complex to simple words [1], while other audiences, such as people with aphasia, will benefit more from short, simple sentences [2]. Although, categorising what complexity means for different audiences is useful for evaluation, TS remains a challenging task to benchmark for the following reasons: 1) The basic concept of **simplicity** (relying on language complexity) is vague and hard to define quantitatively, which means that proficient language users usually come up with different simplifications for a given sentence; 2) The possible usages of TS include scenarios aimed at different **target audiences** (e.g., children, non-native readers, people with

Proceedings of the First Workshop on Current Trends in Text Simplification (CTTS 2021), co-located with SEPLN 2021. September 21st, 2021 (Online). Saggion, H., Štajner, S. and Ferrés, D. (Eds).
laura.vasquezrodriguez@manchester.ac.uk (L. Vásquez-Rodríguez); M.Shardlow@mmu.ac.uk (M. Shardlow); piotr.przybyla@ipipan.waw.pl (P. Przybyła); sophia.ananiadou@manchester.ac.uk (S. Ananiadou)
0000-0002-7313-905X (L. Vásquez-Rodríguez); 0000-0003-1129-2750 (M. Shardlow); 0000-0001-9043-6817 (P. Przybyła); 0000-0001-7116-9338 (S. Ananiadou)

aphasia or dyslexia) and domains (e.g., scientific texts, medical and legal documents), who may require different simplification methods; and 3) Using a gold-standard for TS **evaluation** requires human annotations which is time consuming and costly. This is usually avoided in a way similar to other Natural Language Generation (NLG) tasks (e.g., machine translation) by obtaining human annotated reference simplifications and evaluating systems based on their similarities to these. Although this mechanism of evaluation allows an unlimited number of systems and variants to be evaluated without further human effort, there are a number of factors we have to consider when interpreting the results.

Firstly, there may be many equally good simplifications for a given sentence, so comparison to a single reference may be penalising them unfairly. Despite the existing multiple references in some TS datasets, these cannot capture the rich diversity possible in simplification. Secondly, automatic similarity measures, such as BLEU [3], ROUGE [4] or SARI [5] have been previously shown to have limitations (e.g., weak correlation with human judgement, dependency on quality references, failure to capture task-dependent aspects such as simplicity), both in general tasks [6, 7] and in the context of TS [5, 8, 9]. Thirdly, how data is split between training and test data can influence results. This is well-known in general [10], but has not attracted much attention in TS. Finally, simplification operations may be unevenly distributed in TS datasets, affecting the types of simplifications that a model learns to produce. Test splits may not reflect the same simplification operations as in the training split from the same dataset.

In this paper, we explore the impact of data splits (random and stratified) on English TS datasets and set up a systematic benchmark on the existing datasets with altered distributions. Our contributions are: 1) An operations-based analysis of TS datasets generated by stratification algorithms; 2) A performance evaluation on experimental operation-based datasets; and 3) Recommendations towards a standardised practice for building and evaluating new TS datasets.

2. Related Work

Previous studies [5] have demonstrated the poor-quality of TS datasets used for establishing the state of the art. In particular, Wikipedia-based datasets [11] have incorrectly aligned complex-simple sentence pairs (e.g., sentences with no semantic similarity to each other), and pairs with no simplification or unbalanced simplification operations (e.g., datasets that perform mostly deletions). In contrast, Newsela is a better quality dataset [12] created by professional translators, however it includes a restrictive data agreement that prohibits publishing or sharing data, preventing research reproducibility and the sharing of splits or alignments. Due to these reasons, we have not included Newsela in our study.

Operations-based analysis for datasets is less common and mostly performed based on specific scenarios. Alva-Manchego et al. [13] performed a detailed text features-based analysis in the ASSET dataset, including sentences splits, word deletions, insertions and reorder. Xu et al. [12] analysed a sample of 200 sentences from the PWKP dataset [14] and classified them based on whether they were simplifications or not. Real simplifications were classified under these categories: amount of deletions-only, paraphrasing-only and a combination of both.

Despite the efforts to improve these datasets in terms of the variety of simplification operations performed and the amount of gold-standard references [13], the statistical distributions of these

datasets have not been explored. Recent work from the NLG domain has suggested how the use of random splits can contribute to model performance [15]. Further, there is also a strong argument towards biased or adversarial splits [10], demonstrating that dataset distribution is relevant in NLP. Neither of these has been considered for TS.

Another important fact to consider is the unsuitability of TS evaluation metrics. Over the past few years, the TS research community avoided using the BLEU evaluation metric [8] due to its low correlation with simplicity. Moreover, when simplicity is directly compared with human evaluation, it shows a negative correlation with meaning preservation [16], since building simple sentences also involves removing information from the original ones. As of today, the only available means of TS evaluation is SARI [5], which is not only limited as a measure of 'simplicity gain' in a lexical paraphrasing setting, but also it is potentially flawed when multiple rewrite operations are present [17]. As aforementioned, automatic evaluation of simplicity is still an open question in the TS domain.

For the development of TS systems, simplification operations can also have a fundamental role, where they are explicitly identified or submitted into a TS model. The EditNTS system, a neural programmer-interpreter model [18], detects and predicts ADD, DELETE and KEEP simplification operation during training. Others systems, such as SeqLabel [19], performs an automatic identification of operations in the original parallel corpus, creating a new annotated corpus for training the model.

3. Operation-based Simplification Experiments

We conducted a systematic analysis of the key operations we have identified for all commonly available TS datasets. Initially, we analysed the amount of deletions, insertions and replacements for the different subsets of each TS dataset (i.e., train, development and test when available). We did not include the split operation, since our preliminary analysis using HSplit did not show relevant changes from an edit-distance perspective. Next, we analysed the impact of these operations on the output sentences, comparing how much a complex sentence is changed with the presence of these transformations (Section 3.1 and Section 3.2). Furthermore, we also analysed their distribution, with regards to these simplification operations, proposing new scenarios to benchmark on these new distributions (Section 3.3).

3.1. Creating Operation-based Datasets

We performed our analysis using common Wikipedia-based TS datasets, including: WikiSmall and WikiLarge [11], TurkCorpus [5] and ASSET [13][1]. In particular, we focused on analysing the original TS datasets and our proposed experimental datasets, which are modified versions of WikiLarge and WikiSmall using different distribution methods. We have chosen these resources, since they provide a test, a development and training subset, which are essential for our distribution experiments. We analysed these datasets under the following classifications:

[1]For evaluation, we limited our study to ASSET, since it shows a wider variety of operations based on its edit-distance [20]. Also, due to space constraints, we have included the WikiSmall dataset analysis in the Appendix.

Original distribution: we examined all subsets of the original TS datasets with no modification by applying the metrics defined in section 3.2. We quantified the distribution divergence between subsets (test compared to train and test compared to development) by calculating the Kullback-Lieber (KL-divergence) [21] and Jensen-Shannon divergence (JSD) [22]. As a result, the Wikilarge dataset had a KL-divergence of 0.46 and a JSD divergence of 0.41, confirming that the split of this dataset is not truly random. This can be compared in detail by observing the distribution of these subsets in Figure 1a. Also, we determined that there is a significant amount of sentences with no operations and sentences that have changed 100% during the simplification process. By performing a post-hoc manual inspection of these cases, we noticed that these corresponded to inaccurate simplifications as a product of bad alignments (i.e., poor alignments or noise). Given these results, we proposed additional distributions to improve the distribution of simplification operations in WikiSmall and WikiLarge datasets.

Random distribution: to create randomly distributed datasets, we merged all the subsets from the original dataset into a single dataset, shuffled data using Numpy [23] and recreated the subsets keeping their original size. We repeated this process by using 5 different random seeds (155, 324, 393, 728, 989). The seeds selection was randomly generated, except for 324, which belongs to the original implementation of EditNTS and to the initial explorations in our previous work [20]. In Figure 1, we can see a comparison of the original (Figure 1a) and the random distribution (Figure 1c) for seed 324[2].

Minimised poor-alignments distribution: we manually inspected sentences shown at the right-most of Figure 1a and we observed that sentences close to 100% of change correspond to incorrect simplifications or alignments. Based on this, we created new datasets by removing these poor-alignments ranging from 2% to 20% of sentences with the worst alignment from the original dataset. These splits were not randomised to isolate the effect of removing the poor alignments in TS datasets and duplicates were removed. Figure 1d and 1e show the decrease in the percentage of change in WikiLarge by using this heuristic, including a significantly higher reduction of change in the tests sets compared to the other subsets.

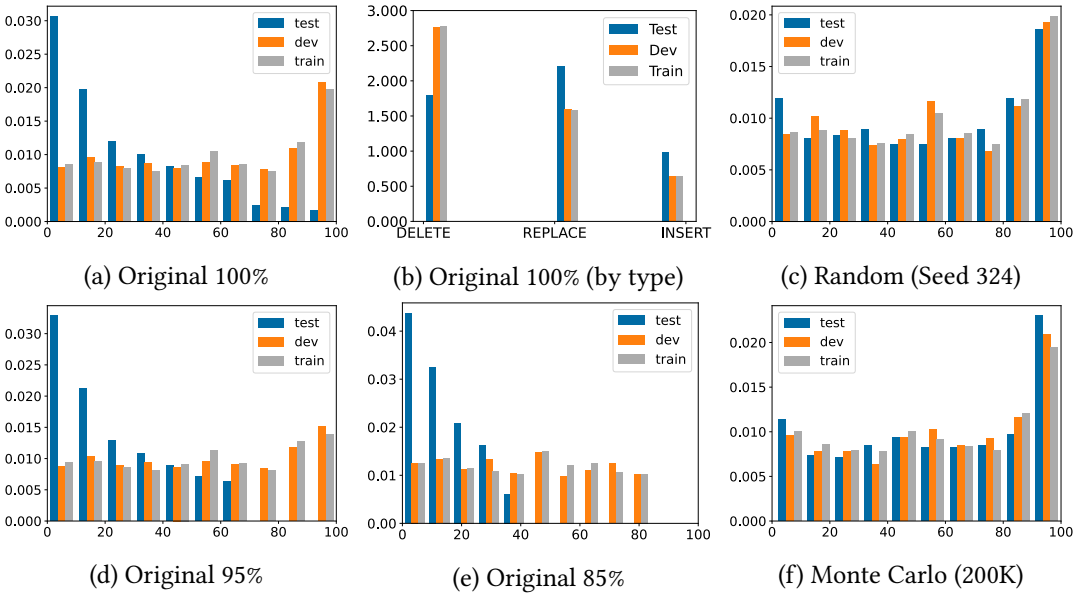
Stratified distribution: sentences in TS datasets can be analysed not only by the changes done from the original to the simplified sentence, but also by the operation type. Our main goal for building new stratified splits is to have similar number of operations of each type (e.g., deletes, inserts and replacements) in each subset. Since a single sentence simplification can involve multiple operations, it is difficult to have the desired distribution between subsets. Among the algorithms evaluated, we selected Monte Carlo Algorithm[3] as our best approach based on the operations distribution. The original datasets were distributed according to this algorithm; datasets subsets were rebuilt and then analysed, likewise to the random distribution. We generated 500,000 random splits searching for one with the best standard deviation between the amount of DELETE, INSERT and REPLACE in each subset. At every 100,000 iterations, we saved the 2 best candidates based on their best standard deviation: one in the training set and

[2]To avoid multiple sources of randomness, we have improved the EditNTS system to guarantee that our model results would be deterministic. Our adaptations to the model can be found at our fork from the original Github https://github.com/lmvasque/EditNTS-eval.

[3]Our implementation of the Monte Carlo Algorithm runs multiple iterations of the random distribution and calculates the standard deviation of each attempt. Finally, it chooses the distribution that has the smallest standard deviation as the best approach for n attempts.

one in the development and test sets, minimising the difference in their individual standard deviation. For WikiLarge, the more suitable splits were iteration 200,000 and 400,000, whereas for WikiSmall these were iterations 300,000 and 500,000. We show the latter in the Appendix.

Once the original and new experimental datasets were created and analysed, we evaluated the effect they had on the performance of the EditNTS [18] model by measuring the change in SARI score when training on the redistributed datasets. We adapted the original code with some minor modifications to run in our setting, including: model randomisation with fixed seeds, scripts for data preprocessing and the automation of test sets evaluation. We trained the models on the original and the experimental subsets (poor-alignments reduction, random and stratified distributions) using the same hyper parameters from the EditNTS model (batch size, epochs, dropout, and learning rate). Next, we evaluated the performance of the newly trained model by using ASSET as an external test subset. Finally, we manually inspected a sample of the model outputs for all the proposed datasets. The adaptations for the EditNTS model, the experimental subsets, the model outputs and the source code are documented via GitHub[4].



(a) Original 100% (b) Original 100% (by type) (c) Random (Seed 324)

(d) Original 95% (e) Original 85% (f) Monte Carlo (200K)

Figure 1: WikiLarge dataset analysis. x-axis: (a), (c) to (f) percentage of change (0% to 100%), (b) operation types; y-axis: (a), (c) to (f) sentence probability density, (b) operation types probability density.

3.2. Quantifying Simplification Operations

Wikipedia-based TS datasets were created collaboratively by volunteers, with the main goal to support learning for non-native speakers. In addition to the rule of writing in Simple English[5], there were no specific guidelines on how to simplify text, such as the type and the amount of simplifications allowed, or whether it should match the original Wikipedia article.

[4]https://github.com/lmvasque/ts-explore
[5]https://simple.wikipedia.org/wiki/Wikipedia:How_to_write_Simple_English_pages

Except for specific studies done by Alva-Manchego et al. [13] and Xu et al. [12], there is no accurate notion of what simplification operations are performed in all TS datasets. These studies are less comprehensive since they investigate specific TS datasets or limited TS operations. Consequently, we analysed common TS datasets by using the following metrics:

Simplification operations count: we quantified the percentage of edits required to transform a complex sentence to a simple one (henceforth, edit-distance [24]). To achieve this, we calculated the edit-distance between two sentences by adapting the Wagner–Fischer algorithm [25] to determine changes from characters-level to a token-level (e.g., words). This method defines how many tokens in the complex sentence were changed in the simplified output (e.g., 2 tokens that were deleted from one version to another is equivalent to 2 changes). Prior to the analysis, sentences were changed to lowercase. Values are expressed as a change percentage, where 0% indicates sentences with no changes and 100% indicates completely different sentences. In Figure 1 we show the edit-distance analysis for WikiLarge, for the original splits (Figure 1a) and also, for the randomised (Figure 1c), poor-alignments-based (Figure 1d, 1e) and stratified splits (Figure 1f). Random and stratified experimental splits clearly show a more even distribution of sentences between subsets, according to the amount of change required to obtain a new simplification from a complex sentence. On the other hand, removing poor-alignments, without a proper distribution leaves the tests sets with the majority of samples with minimal or no change.

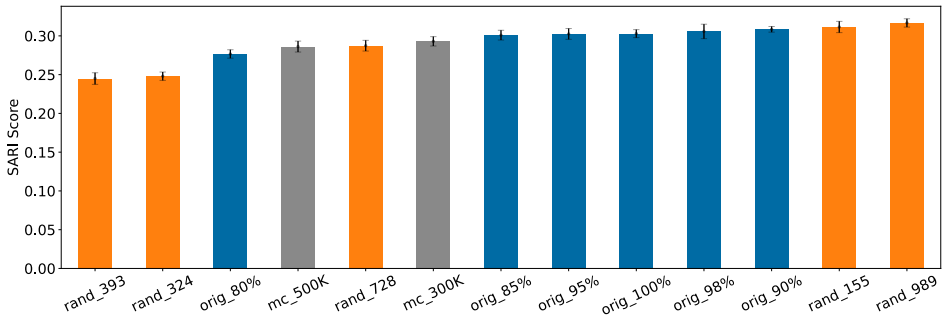
Simplification operations types: after extracting the token-level edits done between two sentences, we classified them into simplification operations: INSERT (a token(s) has been added), DELETE (a token(s) has been removed) and REPLACE (a token(s) has been substituted). These three basic operations can be performed at a lexical-level[6]. We show in Figure 1b the simplification operations types for WikiLarge dataset. These results not only show how unbalanced these operations are between subsets but also the predominance of DELETE operations in the WikiLarge dataset for the development and training subsets. Also, the DELETE effect is also noticeable when we manually checked the outputs of the models. A majority of the simplification operations performed deletions in the original sentence, rather than, performing substitutions or insertions. Furthermore, in Figure 3 we performed a more exhaustive comparison, analysing the operations count and their distribution in all our experiments.

3.3. Evaluating Operation-based Datasets

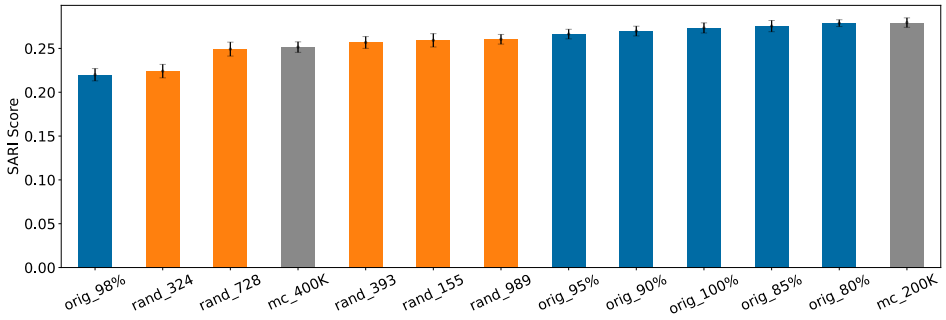
We performed an evaluation of the proposed datasets by retraining the EditNTS model with development and training subsets (for both WikiSmall and WikiLarge)[7]. Once models were trained, we evaluated their performance using the SARI scores provided by the model evaluation scripts. In our evaluation setting, the test subset of the ASSET dataset was used to test the trained models. Also, we reported the average results for the evaluation of all ASSET references in each complex sentence, since our implementation based on EditNTS model evaluated one

[6]We also merged DELETE and INSERT in cases where the same word or phrase is deleted and then inserted again. We called this the MOVE operation. However, since the count of the MOVE operation was insignificant, we only report on three main operations: INSERT, DELETE and REPLACE.

[7]We did not retrain the model using the traditional subsets (i.e. TurkCorpus, ASSET), since our objective was to study the statistical weakness of aforementioned datasets.



(a) WikiSmall



(b) WikiLarge

Figure 2: Comparison of TS models in EditNTS model using ASSET

test reference at a time. Figure 2 shows a comparison between SARI scores in all the models, for WikiLarge-based and WikiSmall-based models. We also include the error bars (standard deviation of ASSET averaged observations) for all calculations. We observed that randomising the distribution and reducing the poor alignments helped for the WikiSmall dataset. Meanwhile, using the Monte Carlo algorithm and performing more substantial reductions in the distribution had a better contribution for WikiLarge.

4. Discussion

Firstly, we conclude from our analyses (Section 3) that the original TS datasets do not follow an even distribution between subsets. We observe that the test, development and train subsets are different when measured by the amount of change from the simple to the complex sentence. Furthermore, our evaluation on these experimental subsets show that the random distributions provide significant variations in SARI scores, even though its composition is similar. For WikiSmall (Figure 2a), the results between random splits showed an increase of up to 7 percentage points in SARI score, just by randomising dataset composition and rebuilding the dataset. For WikiLarge (Figure 2b), we found a similar effect with the Monte Carlo algorithm, which is a split randomised by 200,000 iterations. The main difference is that this algorithm selects the best score among all the generated random samples, rather than any of them. In this setting, the variation of the SARI score is about 5 percentage points. The difference in SARI score should be

interpreted as a measure of simplicity gain, which provides a relative comparison of correctness between simplifications. However, it cannot be interpreted as the best possible simplification, since this evaluation metric fails to measure simplicity alone, as mentioned in Section 2.

Secondly, WikiSmall and WikiLarge datasets show a significant amount of noise and sentences that are not simplifications. Interestingly, we can see in Figure 1e that when we aggressively removed 15% of the dataset, it reduced considerably the amount of sentences with a percentage of change higher than 40%. Despite this, the performance between the original model orig_100% and its reduced version orig_85% did not change more than 0.02% in both WikiSmall and WikiLarge. For the model orig_80% (which has 20% of estimated noise reduction), we observed a different scenario in WikiSmall; since, in comparison with orig_100% model the performance of this dataset dropped 2.6%. WikiSmall dataset is significantly smaller than WikiLarge (3X), and so, such a reduction affects a higher number of real simplifications. In contrast, Wikilarge model orig_98% has a minimal number of noise reduction, keeping its composition almost unchanged. We presume that the decrease in the model performance relates to having the same dataset composition but with less sentence samples (despite their lower quality).

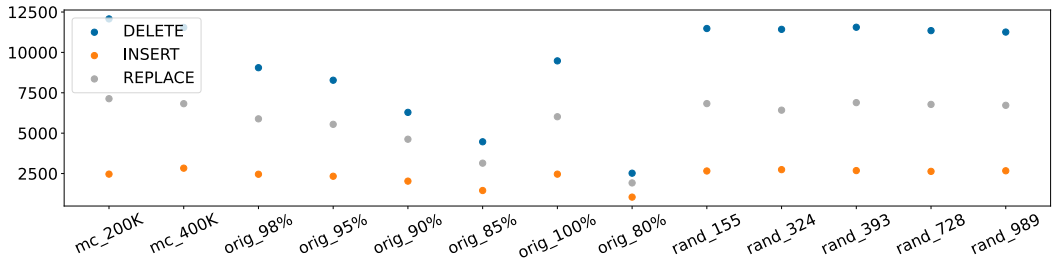


Figure 3: Simplification operations count for WikiLarge Test/Dev subsets

Thirdly, we discuss the datasets composition with respect to the operations count (Figure 3). Due to the large size of the training corpus, the count in the train subset is similar for all the datasets. However, that is not the case of the test and the development subsets, where we noticed meaningful differences. We observe a consistent decrease in all the operations for the models where we removed the 'poor-alignments'. Nevertheless, as we mentioned earlier, orig_80% was the only model which presented a decrease in performance, with a minimal amount of edit operations. On the contrary, despite the similar distribution in operations between random datasets we did observe performance variations between these models. It is relevant to consider that the test and development subsets are quite smaller than the training subset (359 test / 992 dev / 296,402 train in the original sentence pairs). We presume that this could minimise the effect of new distributions towards the model performance. As future work, we would consider changing the original subset sizes to explore further the effect of simplification operations.

5. Recommendations for TS datasets quality assessment

Although the evaluation metrics and model outputs are not globally providing enough information about a dataset, we believe it is important to follow a structured setting to value the

quality of a dataset. To ensure interpretable methods for dataset quality assessment, we make the following recommendations for TS dataset evaluation.

Noisy alignments detection: current TS datasets are automatically aligned, hence, these are likely to have incorrect or unaligned sentence pairs. We propose a heuristic in which these inaccurate alignments can be detected by quantifying the amount of change between the complex sentence and the gold-reference ones. This can be implemented by sorting TS datasets using edit-distance values so sentences with higher amount of changes are grouped together, providing a straight-forward way for detection and removal of noise. The ideal threshold in which sentences are removed can be determined by visually inspecting these groups.

Simplification operations distribution: depending on the audience, some simplification operations can be more useful than others. Ideally, we would expect not only a variety of simplification operations but also, a similar distribution of operations between subsets tailored to a given simplification need. There are valid scenarios in which particular operations could be enough (e.g., REPLACE operation for complex word simplification for non-native speakers). Other areas such as news simplification, require more elaborate constructions which not only involves simplifications at a lexical level, but also at a discourse level (e.g, news for general public targeted for children at schools in the Newsela dataset). By using token-based edit distance, we can perform a global count of simplification operations performed and an evaluation of their distribution as an aid for stratifying TS datasets as needed.

Datasets stability: from our experiments, we have observed that dataset distribution significantly affects TS model performance (measured by an increase or decrease in SARI score). Our recommendation is to perform a dataset randomisation with different random seeds to evaluate the impact of data distribution in TS models performance. In addition, datasets of significant size, such as Wikilarge, showed to be more stable in this setting (less variation in SARI score between random seeds).

6. Conclusions

In this paper, we have performed a systematic analysis of the most common TS operations demonstrating the statistical limitations of English TS datasets. Our analysis can be reproduced through our published scripts, which can also be used to analyse any other TS parallel dataset for quality assessment. Moreover, we carried out a detailed evaluation of all of our experimental settings, including distributions with poor-alignments reduction, randomisation and stratification using the Monte Carlo algorithm. Finally, we have proposed a set of recommendations for the creation of more reliable and standardised datasets for a better environment of TS evaluation resources.

Acknowledgments

We would like to thank Nhung T.H. Nguyen for her valuable discussions and comments. Laura Vásquez-Rodríguez's work was funded by the Kilburn Scholarship from the University of Manchester. Piotr Przybyła's work was supported by the Polish National Agency for Academic Exchange through a Polish Returns grant number PPN/PPO/2018/1/00006.

References

[1] G. H. Paetzold, L. Specia, Unsupervised lexical simplification for non-native speakers, in: 30th AAAI Conference on Artificial Intelligence, AAAI 2016, 2016, p. 3761–3767. URL: http://nlp.stanford.edu/projects/glove/.

[2] J. Carroll, G. Minnen, D. Pearce, Y. Canning, S. Devlin, J. Tait, Simplifying text for language-impaired readers, in: Ninth Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Bergen, Norway, 1999, pp. 269–270. URL: https://www.aclweb.org/anthology/E99-1042.

[3] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: a method for automatic evaluation of machine translation, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 2002, pp. 311–318. URL: https://www.aclweb.org/anthology/P02-1040. doi:10.3115/1073083.1073135.

[4] C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81. URL: https://www.aclweb.org/anthology/W04-1013.

[5] W. Xu, C. Napoles, E. Pavlick, Q. Chen, C. Callison-Burch, Optimizing statistical machine translation for text simplification, Transactions of the Association for Computational Linguistics 4 (2016) 401–415. URL: https://www.aclweb.org/anthology/Q16-1029. doi:10.1162/tacl_a_00107.

[6] K. Ganesan, ROUGE 2.0: Updated and Improved Measures for Evaluation of Summarization Tasks, arXiv (2018). URL: http://arxiv.org/abs/1803.01937. arXiv:1803.01937.

[7] C. Callison-Burch, M. Osborne, P. Koehn, Re-evaluating the role of Bleu in machine translation research, in: 11th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Trento, Italy, 2006. URL: https://www.aclweb.org/anthology/E06-1032.

[8] E. Sulem, O. Abend, A. Rappoport, BLEU is not suitable for the evaluation of text simplification, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 738–744. URL: https://www.aclweb.org/anthology/D18-1081. doi:10.18653/v1/D18-1081.

[9] L. Martin, S. Humeau, P.-E. Mazaré, É. de La Clergerie, A. Bordes, B. Sagot, Reference-less quality estimation of text simplification systems, in: Proceedings of the 1st Workshop on Automatic Text Adaptation (ATA), Association for Computational Linguistics, Tilburg, the Netherlands, 2018, pp. 29–38. URL: https://www.aclweb.org/anthology/W18-7005. doi:10.18653/v1/W18-7005.

[10] A. Søgaard, S. Ebert, J. Bastings, K. Filippova, We need to talk about random splits, arXiv (2020). URL: http://arxiv.org/abs/2005.00636. arXiv:2005.00636.

[11] X. Zhang, M. Lapata, Sentence simplification with deep reinforcement learning, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 584–594. URL: https://www.aclweb.org/anthology/D17-1062. doi:10.18653/v1/D17-1062.

[12] W. Xu, C. Callison-Burch, C. Napoles, Problems in current text simplification research: New data can help, Transactions of the Association for Computational Linguistics 3 (2015)

283–297. URL: https://www.aclweb.org/anthology/Q15-1021. doi:10.1162/tacl_a_00139.

[13] F. Alva-Manchego, L. Martin, A. Bordes, C. Scarton, B. Sagot, L. Specia, ASSET: A Dataset for Tuning and Evaluation of Sentence Simplification Models with Multiple Rewriting Transformations, arXiv (2020). URL: http://arxiv.org/abs/2005.00481. doi:10.18653/v1/2020.acl-main.424. arXiv:2005.00481.

[14] Z. Zhu, D. Bernhard, I. Gurevych, A monolingual tree-based translation model for sentence simplification, in: Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), Coling 2010 Organizing Committee, Beijing, China, 2010, pp. 1353–1361. URL: https://www.aclweb.org/anthology/C10-1152.

[15] K. Gorman, S. Bedrick, We need to talk about standard splits, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 2786–2791. URL: https://www.aclweb.org/anthology/P19-1267. doi:10.18653/v1/P19-1267.

[16] M. Schwarzer, D. Kauchak, Human Evaluation for Text Simplification: The Simplicity-Adequacy Tradeoff, Technical Report, SoCal NLP Symposium, 2018.

[17] F. Alva-Manchego, L. Martin, A. Bordes, C. Scarton, B. Sagot, L. Specia, ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 4668–4679. URL: https://www.aclweb.org/anthology/2020.acl-main.424. doi:10.18653/v1/2020.acl-main.424.

[18] Y. Dong, Z. Li, M. Rezagholizadeh, J. C. K. Cheung, EditNTS: An neural programmer-interpreter model for sentence simplification through explicit editing, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 3393–3402. URL: https://www.aclweb.org/anthology/P19-1331. doi:10.18653/v1/P19-1331.

[19] F. Alva-Manchego, J. Bingel, G. Paetzold, C. Scarton, L. Specia, Learning how to simplify from explicit labeling of complex-simplified text pairs, in: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Asian Federation of Natural Language Processing, Taipei, Taiwan, 2017, pp. 295–305. URL: https://www.aclweb.org/anthology/I17-1030.

[20] L. Vásquez-Rodríguez, M. Shardlow, P. Przybyła, S. Ananiadou, Investigating text simplification evaluation, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, Online, 2021, pp. 876–882. URL: https://aclanthology.org/2021.findings-acl.77. doi:10.18653/v1/2021.findings-acl.77.

[21] S. Kullback, R. A. Leibler, On Information and Sufficiency, The Annals of Mathematical Statistics 22 (1951) 79–86. doi:10.1214/aoms/1177729694.

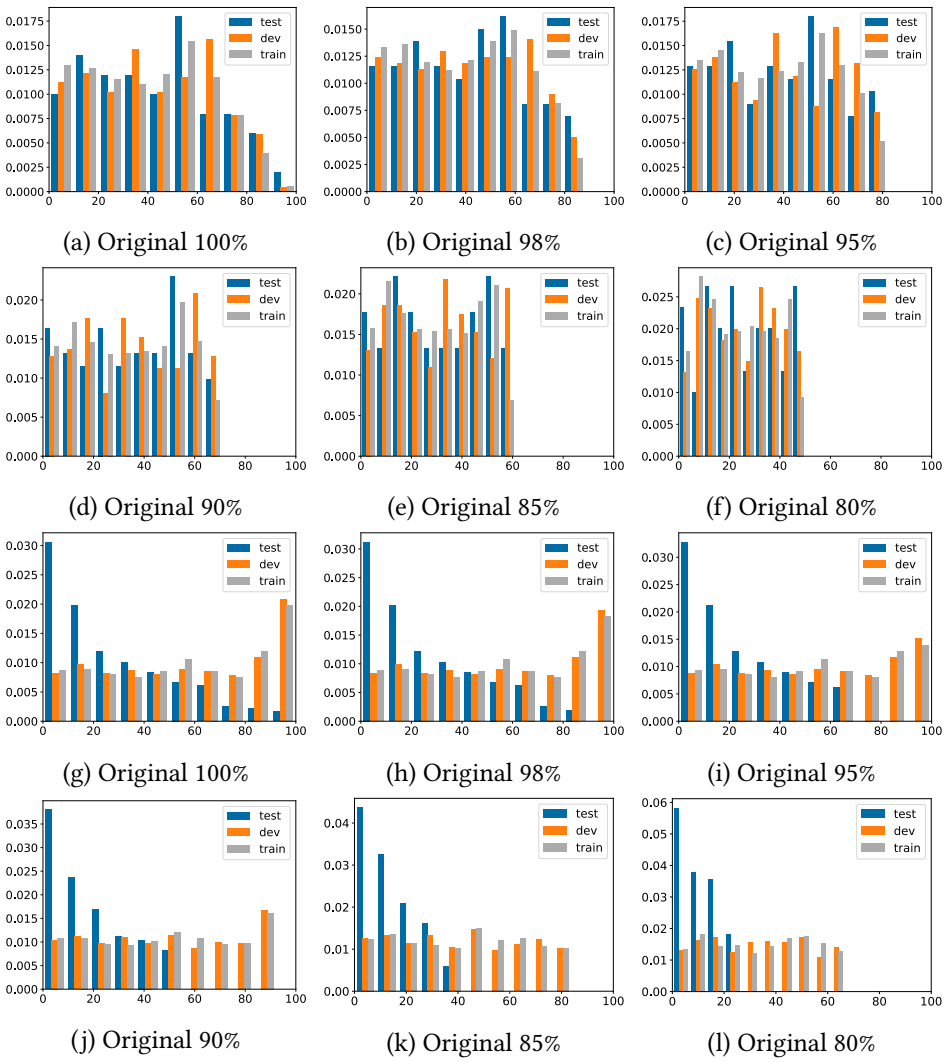
[22] J. Lin, Divergence Measures Based on the Shannon Entropy, Technical Report, IEEE, Transactions on Information Theory, 1991.

[23] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, T. E. Oliphant, Array programming with NumPy, Nature 585 (2020) 357–362. URL: https://doi.org/10.1038/s41586-020-2649-2.

doi:10.1038/s41586-020-2649-2.
[24] V. Levenshtein, Binary codes capable of correcting deletions, insertions, and reversals, Soviet Physics Doklady 10 (1966).
[25] R. A. Wagner, M. J. Fischer, The String-to-String Correction Problem, Journal of the ACM (JACM) 21 (1974) 168–173. URL: https://dl.acm.org/doi/10.1145/321796.321811. doi:10.1145/321796.321811.

A. Appendices

A.1. Poor-alignments analysis



(a) Original 100% (b) Original 98% (c) Original 95%

(d) Original 90% (e) Original 85% (f) Original 80%

(g) Original 100% (h) Original 98% (i) Original 95%

(j) Original 90% (k) Original 85% (l) Original 80%

Figure 4: Poor-alignments dataset analysis for WikiSmall (a-f) and WikiLarge (g-l)

A.2. Random-based analysis

(a) Random (Seed 155) (b) Random (Seed 324) (c) Random (Seed 393)

(d) Random (Seed 728) (e) Random (Seed 989) (f) Monte Carlo (300K)

(g) Monte Carlo (500K) (h) Random (Seed 155) (i) Random (Seed 324)

(j) Random (Seed 393) (k) Random (Seed 728) (l) Random (Seed 989)

(m) Monte Carlo (200K) (n) Monte Carlo (400K)

Figure 5: Random-based dataset analysis for WikiSmall (a-g) and WikiLarge (h-n)