SiTSE: Sinhala Text Simplification Dataset and Evaluation

SURANGIKA RANATHUNGA, Massey University, Auckland, New Zealand
RUMESH SIRITHUNGA, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka
HIMASHI RATHNAYAKE, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka
LAHIRU DE SILVA, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka
THAMINDU ALUTHWALA, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka
SAMAN PERAMUNA, Sabaragamuwa University of Sri Lanka, Belihuloya, Sri Lanka
RAVI SHEKHAR, University of Essex, Colchester, United Kingdom of Great Britain and Northern Ireland

Text Simplification is a task that has been minimally explored for low-resource languages. Consequently, there are only a few manually curated datasets. In this article, we present a human-curated sentence-level text simplification dataset for the Sinhala language. Our evaluation dataset contains 1,000 complex sentences and 3,000 corresponding simplified sentences produced by three different human annotators. We model the text simplification task as a zero-shot and zero-resource sequence-to-sequence (seq-seq) task on the multilingual language models mT5 and mBART. We exploit auxiliary data from related seq-seq tasks and explore the possibility of using intermediate task transfer learning (ITTL). Our analysis shows that ITTL outperforms the previously proposed zero-resource methods for text simplification. Our findings also highlight the challenges in evaluating text simplification systems and support the calls for improved metrics for measuring the quality of automated text simplification systems that would suit low-resource languages as well. Our code and data are publicly available: https://github.com/brainsharks-fyp17/Sinhala-Text-Simplification-Dataset-and-Evaluation.

CCS Concepts: • **Computing methodologies** → **Artificial intelligence**; **Natural language processing**; **Language resources;**

Data creation was funded by a Senate Research Committee grant of the University of Moratuwa. Training of models in this work was made possible by the AWS cloud credits received by Surangika Ranathunga under the *AWS Cloud Credit for Research* program.

Authors' Contact Information: Surangika Ranathunga, Massey University, Auckland, New Zealand; e-mail: s.ranathunga@massey.ac.nz; Rumesh Sirithunga, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka; e-mail: rumeshmadhusanka.17@cse.mrt.ac.lk; Himashi Rathnayake, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka; e-mail: himashi.17@cse.mrt.ac.lk; Lahiru de Silva, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka; e-mail: lahirudesilva.17@cse.mrt.ac.lk; Thamindu Aluthwala, Department of Computer Science and Engineering, University of Moratuwa, Moratuwa, Sri Lanka; e-mail: thamindu.17@cse.mrt.ac.lk; Saman Peramuna, Sabaragamuwa University of Sri Lanka, Belihuloya, Sri Lanka; e-mail: dmsperamuna@gmail.com; Ravi Shekhar, University of Essex, Colchester, United Kingdom of Great Britain and Northern Ireland; e-mail: r.shekhar@essex.ac.uk.

ACM 2375-4699/2025/05-ART51
https://doi.org/10.1145/3723160

Additional Key Words and Phrases: Low-resource languages, sequence-sequence pre-trained language models, mBART, mT5, transfer learning, intermediate task transfer learning

ACM Reference Format:
Surangika Ranathunga, Rumesh Sirithunga, Himashi Rathnayake, Lahiru de Silva, Thamindu Aluthwala, Saman Peramuna, and Ravi Shekhar. 2025. SiTSE: Sinhala Text Simplification Dataset and Evaluation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 24, 5, Article 51 (May 2025), 19 pages. https://doi.org/10.1145/3723160

1 Introduction

Text Simplification (TS) refers to modifying a given text while preserving its original meaning to improve its readability and understanding [4]. TS has proven beneficial in improving comprehension in low-literacy readers [48], helping in learning difficulties [66], teaching L2 learners [19], and communicating domain-specific information such as medical text to laypersons [28]. Thus, TS increases the fairness of textual information processing systems [28]. TS can also be used to improve the performance of other **Natural Language Processing (NLP)** tasks such as Machine Translation [1] and Summarization [72].

Despite their many benefits, TS systems have been implemented mainly for high-resource languages such as English, French, and Spanish. TS research on low-resource languages is limited, except for some work on Urdu, Bengali, Marathi, Basque, and Slovene [6, 31, 51]. The main reason for this is due to the lack of complex–simple word lexicons and/or semantic/syntactic parsers used for the early TS systems. Even with the recent focus on low-resource NLP [71], very little attention is paid to the TS task due to the unavailability of datasets and models. Creating high-quality and large TS datasets[1] is costly and time-consuming since humans have to read, comprehend, and identify possible simplification points and relevant operations and finally carry out the simplification. Some datasets were automatically curated using Wikipedia to reduce dataset creation costs, but these datasets have not been very useful for TS [3]. Thus, human-curated TS datasets still require language-specific attention.

In this article, we focus on Sinhala TS. Sinhala is an Indo-Aryan language with its own script and alphabet used in the island nation of Sri Lanka, spoken by about 22 million people. Sinhala is a low-resource language due to the scarcity of digital language resources and lack of collaborative research [20, 64]. We present the first-ever sentence-level **Sinhala Text Simplification Evaluation (SiTSE)** dataset. Our dataset consists of 1,000 complex sentences from Sinhala official government documents. Each complex sentence has three simple references produced by Sinhala language experts, thus resulting in 3,000 sentence pairs.

The state-of-the-art solutions for TS are based on Deep Learning techniques, in particular **sequence–sequence pre-trained Language Models (sqPLMs)** [31, 47]. Semi- and unsupervised neural methods have been used to tackle the data scarcity problem. However, the existing semi- and unsupervised neural TS systems [77, 94] have mainly been tested using **high-resource languages (HRLs)** in a simulated low-resource setting. For example, Surya et al. [77], Zhao et al. [94] used the English Wikipedia; Martin et al. [47] used the CommonCrawl of English, French, and Spanish; and Mallinson et al. [45] used 6M parallel sentences for English–German.

Sinhala has a fraction of these monolingual and multilingual data. On the positive side, Sinhala has been included in sqPLMs such as mBART [41] and mT5 [88]. Despite Sinhala being underrepresented in these models, their use has shown very promising results for related sequence–sequence tasks such as Translation and auto-regressive text generation in the context of Sinhala

[1] A parallel corpus, where each entry resembles a complex sentence and its simple version.

[38, 57, 79]. Sinhala also has small amounts of cleaned parallel data [26], as well as relatively large amounts of web-mined noisy parallel data [65].

We experiment with sqPLM-based unsupervised TS models for Sinhala, with our SiTSE dataset as the test set. Our TS models are based on the concept of **intermediate task transfer learning (ITTL)**. In addition to using a single task for the intermediate task as reported in previous research [60, 61], we fine-tune the sqPLM sequentially with a set of auxiliary tasks. Given that the type of intermediate task matters [60], all these auxiliary tasks are sequence–sequence tasks–Sinhala paraphrasing, Translation, and English simplification.

All our ITTL models report SARI scores comparable to those reported for English datasets [4], as well as high BERTScores. The best SARI result was obtained from the sequential fine-tuning setup with auxiliary task data. Thus, we present our work as a case study on how low-resource languages can exploit existing data and models to achieve acceptable performances on new tasks even without task-specific training data.

We also conducted two human evaluations on the results and the manually simplified datasets to shed further light on the model performance and data quality. In one evaluation, three evaluators scored the simplifications with respect to fluency, adequacy, and simplicity on a Likert scale of 1 to 5. In the other evaluation, three separate evaluators recorded the frequency of simplification-related errors. Both human analyses confirmed that our models outperform the existing sqPLM-based models for text simplification. Our comprehensive analysis also introduced two new criteria for the human evaluation of text simplification models. We also show that determining the best model with respect to all error analysis criteria is challenging.

2 Related Work

2.1 Text Simplification Datasets

To train as well as to evaluate text simplification models, a parallel dataset is required where each parallel pair corresponds to a complex sentence/document and its simpler version. Early research on English text simplification used automatically aligned sentences from the complex and simple Wikipedias [53]. PWKP [95], C&K-2 [32], Wikilarge [90], and AlignedWL and RevisionWL [85] are example corpora automatically compiled from Wikipedia. However, the suitability of such automatically aligned datasets for text simplification has been criticized [4]. The TurkCorpus [87] was derived by manually simplifying a subset of the PWKP corpus. Simplification was conducted mainly via lexical paraphrasing. ASSET provides 10 simplifications per sentence for a subset of TurkCorpus [3]. The HSplit corpus was created by simplifying the test set of the TurkCorpus [75]. Here, simplification was mainly achieved by sentence splitting.

The SimPA and the Newsela [69, 86] corpora are manually compiled datasets for English. Newsela [86] is an example of a document-level dataset (for English and Spanish). News articles were manually simplified up to five levels by professionals considering children in different educational levels. OneStopEnglish [81] is another document-level corpus for English TS.

Table 1 lists some of the available datasets for non-English languages. We note that most of the datasets are aligned at the sentence level, and manually aligned datasets have sizes of around 1,000. Furthermore, most datasets provide only one reference per complex sentence, which indicates the difficulty of creating large-scale human-curated text simplification datasets.

2.2 Pre-trained Language Models

The aim of neural language modeling is to learn a distributed representation over the linguistic units (e.g., words, sentences) of a language [12]. Neural language models came into prominence with the success of word representation models such as Word2Vec [49]. Later, with the introduction of the Transformer architecture [82], neural language models learned from billions of text tokens

Table 1. Existing Non-English Text Simplification Datasets

Paper	Language	Alignment	Simple Dataset Size	# of Refs	Source	Compilation
Goto et al. [29]	Japanese	sentence	13,336	1	news	Instructors simplified full articles and sentences were manually aligned
Bott and Saggion [13]	Spanish	sentence	590	1	news	Manually simplify articles and sentences were automatically aligned
Klaper et al. [34]	German	sentence	7,000	1	websites	Start with complex–simple document pairs and automatically aligned sentences
Tonelli et al. [80]	Italian	sentence	1,166	1	wikipedia	By analyzing page edits, complex–simple pairs were extracted and later manually curated
Brunato et al. [16]	Italian	sentence	1,036	1	childrens novel	Used novels and their manually simplified version, and manually extracted parallel sentences
Brunato et al. [15]	Italian	sentence	63,000	1	open	Automatically web-mined
Xu et al. [86]	Spanish	document	1,221	5	news	Manually simplified by professionals
Gala et al. [27]	French	document	79	4	literary, scientific	Manually simplified by professionals
Anees and Rauf [6]	Urdu	sentence	610	2	literary	Automatically simplified
Caseli et al. [17]	Brazilian Portuguese	sentence	2,116	1	news	Manually aligned
Klerke and Søgaard [35]	Danish	sentence	48,186	1	news	Automatically aligned
Brouwers et al. [14]	French	sentence	235	1	Wikipedia, literature	First parallel articles were identified, and sentence alignment was done semi-automatically
Säuberli et al. [68]	German	sentence	3,666	2	news	Start with simple–complex pair of documents and automatically align sentences in training set and manually align validation and test sets
Specia [73]	Portuguese	sentence	4,483	1	news	Manual simplification
Battisti et al. [11]	German	document	8,400	1	news	Automatically simplified
Miliani et al. [50]	Italian	sentence	736	-	government	Manual simplification
Mondal et al. [51]	Bengali, Marathi	sentence	500	1	news	Manual simplification
Mondal et al. [51]	Hindi	sentence	1,041	1	news, Wikipedia	Manual simplification
Stodden et al. [74]	German	document	1,239	1	news, general	Manual and automatic

came into play, and these are referred to as **pre-trained Language Models (PLMs)** or Foundation Models, among other terms. Based on their training, PLMs can be categorized as encoder-only, decoder-only, and encoder–decoder models. Models such as BERT [21] and RoBERTA [40] are examples of encoder-only models. Models such as GPT [62] and even recently introduced models such as ChatGPT and Llama [25] (which are commonly referred to as **Large Language Models**

(LLMs)) are examples of decoder-only models. BART [39] and T5 [63] are encoder–decoder models, and mBART [41] and mT5 [88] (respectively) are their multilingual versions. We call these sqPLMs. Such models have been trained in a seq-seq manner and are suitable for tasks such as TS, which expects to generate a new text string conditioned on an input text string.

2.3 Neural Text Simplification

The nature of the TS task makes the encoder–decoder neural architectures a natural solution, where the simplification task is modeled as a monolingual translation task [56, 83]. However, vanilla encoder–decoder models tend to miss transformations needed for simplification [91].

One way to mitigate this problem is to combine the neural model with a rule-based system [42, 76, 91, 92]. Another solution is to add artificial tokens to the source text to control the simplification process at the decoder [44, 46, 55, 70]. Nakamachi et al. [52], Zhang and Lapata [90] combined reinforcement learning with the neural model. Kriz et al. [36] added a complexity-weighted loss function to encourage the model to choose simpler words. Several research works exploited the knowledge of related tasks such as entailment, translation, and paraphrasing in a multi-task setup [1, 30]. Another line of research incorporated multiple neural models to separately predict the simplifying operation and execute that edit operation [2, 24]. Bahrainian et al. [10] employed a secondary neural model to translate complex terms to their simpler versions.

To implement TS systems for languages that have no parallel simplification data, three solutions are available: (1) creation of simplification data by data augmentation [7, 47], (2) zero-shot text simplification on a multi-tasking sequence–sequence model that exploits TS data of HRLs [45], and (3) auto-encoders [77, 94].

More recent research has used sqPLMs as well as decoder-only PLMs for TS. Kew et al. [33] carried out a comprehensive evaluation across 44 such models and concluded that prompting on decoder-only PLMs (i.e., LLMs) is the best. However, they only considered English datasets.

Similar to us, Martin et al. [47], Ryan et al. [67] used sqPLMs for unsupervised TS. However, Martin et al. [47] simply fine-tuned mBART with web-mined paraphrases for the target language. Similar to us, Ryan et al. [67] used a TS dataset from another language to fine-tune the sqPLM, but did not consider ITTL.

2.4 Intermediate Task Transfer Learning

ITTL, or **Supplementary Training on Intermediate Labeled data Tasks (STILTs),** means fine-tuning a PLM with an auxiliary task before fine-tuning the same with the target task [61]. This intermediate task can either be a single task or a multi-task setup [61].

In the context of encoder-based pre-trained models, Phang et al. [60] used ITTL in a cross-lingual zero-shot setup. Artetxe et al. [8] extensively experimented with the impact of translated data on ITTL. They showed that rather than using English task data as an intermediate task, using its back-translated version performs better. As for sequence–sequence models, Takeshita et al. [78] used English monolingual summarization and machine translation as intermediate tasks for multilingual text summarization. For Neural Machine Translation, Nayak et al. [54] first fine-tuned an sqPLM with auxiliary domain data before fine-tuning the same with target domain data. In the context of Sinhala, Dhananjaya et al. [23] employed ITTL for sentiment analysis.

3 Sinhala Text Simplification Evaluation Dataset

Addressing the unavailability of a Sinhala text simplification dataset, we present the **SiTSE** (**Si**nhala **T**ext **S**implification **E**valuation) dataset, an evaluation dataset for the Sinhala language composed of 1,000 complex sentences, each with three simplified reference sentences (thus 3,000 complex–simple pairs).

Complex sentence:
එක් එක් පළාත සඳහා ඇවැසි වැඩසටහන් සංවිධානය කර ව්‍යාපෘතිය දැනුවත් කරන ලෙස පළාත් නිලධාරීන්ගට දැනුම් දී තිබූ නමුත් එම කාර්යය නිසියාකාරව සිදු කිරීමට අපොහොසත් වූ පසුබිමක් තුළ අපේක්ෂිත ප්‍රමාණයන් වැඩසටහන් පවත්වා ගැනීමට අපහසු වී තිබෙ .
(Provincial officials were instructed to organize the required programs for each province and make them aware of the project, but in the background of failing to carry out that task properly, it has become difficult to maintain the desired size of programs.)

Simplification 1:
එක් එක් පළාත සඳහා වැඩසටහන් සංවිධානය කර ව්‍යාපෘති ගැන ජනතාව දැනුවත් කරන ලෙස නිලධාරින්ට උපදෙස් ලැබී ඇත. නමුත් එම කාර්යය නිවැරදිව සිදු වී නැත. එම නිසා අපේක්ෂිත වැඩසටහන් සංඛ්‍යාව සංවිධානය කිරීම අපොහොසත් වී ඇත .
(Officials have been instructed to organize programs for each province and make the public aware of the projects. But that work has not been done correctly. Therefore, the expected number of programs failed to organize.)

Simplification 2:
පළාත් නිලධාරීන්ට අදාල පළාත් සඳහා ව්‍යාපෘති සංවිධානය කර දැනුවත් කරන ලෙස උපදෙස් ලැබී තිබුණ ද නිලධාරීන් එයට අපොහොසත් වී ඇත. මේ පසුබිමේ නිවැරදි වැඩසටහන් සංඛ්‍යාවක් පවත්වා ගැනීමට අපහසුය .
(Although the provincial officials have been instructed to organize and inform the relevant provinces about the projects, the officials have failed to do so. Against this background it is difficult to maintain an accurate number of programs.)

Simplification 3:
එක් එක් පළාත සඳහා ඇවැසි වැඩසටහන් සංවිධානය කර ව්‍යාපෘතිය දැනුවත් කරන ලෙස පළාත් නිලධාරීන්ගට දැනුම් දී ඇත. නමුත් එම කාර්යය නිවැරදිව සිදු කිරීමට නොහැකි වූ පසුබිමක් තුළ අපේක්ෂිත වැඩසටහන් ගණන පවත්වා ගැනීමට අපහසු වී ඇත .
(Provincial officials have been instructed to organize the required programs for each province and make them aware of the project. But it has been difficult to maintain the desired number of programs in a background where the task could not be done correctly.)

Fig. 1. A sample sentence from the SiTSE dataset with English translations.

Data Selection: Our dataset is from the Sinhala side of the human-created English–Sinhala–Tamil parallel corpus [26], which belongs to the government domain. In Sinhala, government documents are written in the *written* form of the language, whereas Sinhala speakers use the much simpler *spoken* version of the language for their day-to-day communication. These documents heavily use the government-approved standardized term glossaries that contain words that are rarely used in the daily usage of Sinhala speakers. The sentences are long with an average sentence length of 34.98 words. Three of the authors manually reviewed the corpus to select complex sentences that contain rare words and large sentence lengths.

Annotation Process: Three human participants performed the simplification.[2] Following Alva-Manchego et al. [3], annotators were instructed to perform the following operations to simplify text:

— Extract the main idea of the sentence.
— Split long sentences into shorter ones.
— Perform lexical reordering, replacing complex words with commonly used simple words.

Annotators were provided three rounds of pilots. In each round, three of the authors went over the complex–simple sentence pairs and provided feedback to the annotators. The annotation was performed in batches, and we monitored each batch to ensure the quality. In Figure 1, we present an example of simplifications from the SiTSE dataset, having both complex and corresponding simplified sentences. We could see that annotators performed different operations for the simplification.

Dataset Statistics: For each complex sentence in the dataset, there are three simplifications. Thus, for the 1,000 complex sentences, there are 3,000 simple sentences. On the contrary, we note that most of the datasets reported in Table 1 have only one simplification per complex sentence. In

[2]Details of human participants are in Appendix C.

Table 2. Statistics of the SiTSE Dataset

Property	Simple 1	Simple 2	Simple 3
Avg. Sentence Length	11.23	11.92	14.39
Additions	0.23	0.13	0.09
Deletions	0.16	0.13	0.07
Sentence splits	1.96	1.51	1.33
Compression ratio	1.03	1.00	1.02

Amount of additions, deletions, splits is calculated by taking ratio of those between original and reference sentence. The compression ratio is calculated by dividing the number of characters in the reference sentence by the number of characters in the original sentence.

Table 2, we present the statistics of the SiTSE dataset, averaged across all the annotators.[3] We note that the annotators have carried out simplification mainly by splitting long sentences. Thus, most of the simplified versions have multiple sentences and our dataset can be considered similar to the HSplit dataset [75]. On average, the annotators have produced shorter sentences, with an 11 to 14 average words per sentence, whereas the original sentence has an average length of 34.98 words. We provide a human analysis of this dataset in Section 6.

4 Models

4.1 Baseline Models

We used three baseline models based on sqPLMs, as well as a pivot-based model.

Sequence–sequence Pre-trained Multilingual Models (sqPLMs): Pre-trained mBART and mT5 models were used to generate Sinhala sentences without any explicit fine-tuning for text simplification. Here we input complex sentences into mBART and mT5 and get the output sentences. This is similar to the random baseline.

Zero-shot Text Simplification (Si Zero-shot TS):. The goal is to test the model's performance when there is no data for the target language but other language data is available. First, we fine-tuned sqPLMs with an HRL TS dataset. We then tested it for Sinhala TS. Note that this is the zero-shot method used by Ryan et al. [67].

Text Simplification Using Paraphrase Mining (TS-Mining): This is the unsupervised TS system proposed by Martin et al. [47]. Similar to Martin et al. [47], first, we mined the paraphrases using LASER embeddings [9] for Sinhala sentences in the Si-CC15 monolingual dataset [22]. Then, we used these paraphrases along with controllable sentence simplification [46] to fine-tune a sqPLM for Sinhala TS.

Pivot-based Text Simplification (Pivot): Pivoting has been reported as a strong baseline in TS [47]. Here, we assume that the availability of a system for simplification in another language and translation from Sinhala is possible. Specifically, we used Google Translate[4] to translate Sinhala sentences to English and then used the model of Martin et al. [47] for English simplification. Finally, the simplified English sentences are translated back into Sinhala.

4.2 ITTL-based Models

In our case, although we now have a human-curated dataset to evaluate Sinhala TS systems, we do not have parallel data to train a model for the same. This is a classic example of a zero-shot problem.

[3]We used the implementation in https://github.com/feralvam/easse for the calculation.
[4]https://translate.google.com/

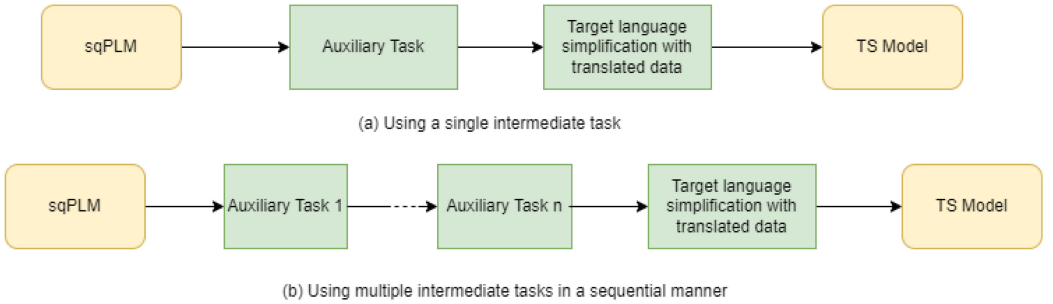

Fig. 2. ITTL with (a) a single intermediate task and (b) a sequence of intermediate tasks.

We cast this as a zero-resource text simplification problem by translating a text simplification corpus of an HRL into the Sinhala language, thus creating a synthetic Sinhala simplification dataset. The translated corpus was used to fine-tune the sqPLM. We refer to this as the *Si-simp* model.

Then we extended this model with ITTL. For the auxiliary tasks, we consider the following sequence–sequence tasks:

— Translation (*Trans*): Fine-tune the sqPLM using a parallel corpus used for Machine Translation with Sinhala on the target side. We hypothesize this task would help the decoder of the sqPLM to generate better Sinhala sentences with good fluency since the decoder sees more Sinhala sentences during fine-tuning.
— Paraphrasing (*Para*): Fine-tune the sqPLM with paraphrases of the considered language. We used the web mining technique of Martin et al. [47] to generate paraphrases automatically. The paraphrasing task helps the model preserve meaning.
— HRL text simplification (*En-simp*): Fine-tune the sqPLM with a text simplification corpus of an HRL similar to our *Si Zero-shot TS* baseline. Fine-tuning with a dataset of the same task teaches task characteristics to the model. As discussed in Section 2.4, this is a common choice for the intermediate task.

For sequential task fine-tuning, we used combinations of these three auxiliary tasks to determine which task(s)/ordering has the highest positive impact on the Sinhala TS. These different modes of ITTL are shown in Figure 2. While it is possible to experiment with other auxiliary tasks, we are not aware of any other sequence–sequence datasets for Sinhala.

5 Experiment Setup

For the translation task, we used the SiTa corpus with 56,000 parallel Sinhala–Tamil–English sentence pairs extracted from official government documents of Sri Lanka [26]. For the paraphrasing task, we generated 7,000 paraphrased sentence pairs from a portion of 600,000 sentences of the Si-CC15 dataset using the method of Martin et al. [47] for paraphrase mining. We limit ourselves to 600,000 sentences from the Si-CC15 dataset due to computational resource limitations. When using this method, we used the Newsela [86] dataset for the English text simplification task as the HRL TS dataset and used Google Translate to translate the Newsela corpus to Sinhala.

We used mBART50 and mT5-base models for our experiments. Fairseq [58] was used for fine-tuning mBART models, and HuggingFace Transformer [84] was used for fine-tuning mT5 models. Details on the hyper-parameters and the computation resources are in Appendix D.

5.1 Human Evaluation

We employ human evaluators to measure the simplification capability of humans (see Section 3) as well as the TS models using *adequacy* (output sentence preserving the main idea of the

complex sentence), *fluency* (output sentence free from grammatical errors), and *simplicity* (output sentence easier to understand than the complex sentence). These are the commonly used criteria for human evaluation on TS [42, 47]. We randomly selected 50 sentences from the human-created as well as the model-generated datasets. Three human participants rated the simplified sentence compared to the complex sentence for each of the three criteria and gave a score on a Likert scale of 1 to 5. We calculated the average for each experiment across 50 sentences under each dimension.

In order to further understand the difference in the simplification capabilities of humans and the models, we performed a separate error analysis by using 50 randomly selected sentences and three separate evaluators. We used the of Maddela et al. [42] for this purpose. The error categories are fluency errors, hallucinations, anaphora resolution errors, and bad substitutions. In addition, we added two new error categories: near-exact copy and missing major fact. This is because there can be an output sentence that misses a major fact or is the same as the complex sentence while not having any of the other errors. These two error categories are mutually exclusive. Further details of these error categories are in Appendix B.

6 Results and Discussion

6.1 Quantitative Results

There are no agreed-upon evaluation metrics for text simplification [3]. Following the suggestion of Alva-Manchego et al. [5], we first computed the BERTScore [89] to measure the quality of the system output and then used SARI to measure the simplicity. The higher the SARI and BERTScore, the better the simplification is. More details of these metrics are in Appendix A. We do not use N-gram-based metrics such as BLEU [59] because they are not suitable for evaluating text simplification [75]. Recently, several learnable TS metrics have been proposed [18, 37, 43, 93]. However, these metrics are based on a neural model trained using human-annotated data, which is not available for low-resource languages such as Sinhala.

Results are reported in Table 3. First and foremost, we can observe that BERTScore is comparatively higher (more than 81) for most of the models, except for *Si Zero-Shot-TS* and the *Pivot-based* baselines. As observed by Alva-Manchego et al. [5], a higher BERTScore means good output sentence quality. This means that most models provide good-quality sentences–simple or not. Manual inspection showed that the output of the *Pivot-based model*[5] and *Zero-Shot-TS*[6] baselines is of very bad quality. Therefore, we did not use these two baseline models in further analysis. The rest of the models are analyzed using SARI scores.

All our ITTL models have comparable results for both mBART and mT5, with mT5 being slightly better. The best result is reported by the *Trans→En-simp→Si-simp* ITTL strategy on mT5. We also note that these results are comparable to those reported by Alva-Manchego et al. [4] for English text simplification. Moreover, all our models are noticeably better than the model of Martin et al. [46] (*TS-mining*).

ITTL that uses translation as the auxiliary task shows the best performance. Similarly, when combining auxiliary tasks, starting with the translation task gives the best result. Thus, our observations align with those of Takeshita et al. [78]. Combining all three tasks sequentially does not lead to noticeable gains. We also note that SARI scores can sometimes be misleading. For example, the amount of deletions reported in the *TS-mining* baseline is lower than many other models. However, it has the shortest sentence length, which suggests that more content of the original sentence has been removed, which contradicts the corresponding value reported by SARI.

[5]Adds out-of-context words.
[6]Generates code-mixed data.

Table 3. Quantitative Results

Experiments	Avg. Sent Length	mBART					mT5				
		SARI	Components of SARI			BERT	SARI	Components of SARI			BERT
			F_{ADD}	F_{DEL}	F_{KEPT}			F_{ADD}	F_{DEL}	F_{KEPT}	
sqPLMs (Baseline)	39.12	28.80	0.19	7.40	78.81	86.20	16.35	0.08	47.00	1.99	15.21
Si Zero-shot-TS (Baseline)	34.82	17.32	0.40	47.00	4.56	31.26	16.83	0.36	46.99	3.15	25.10
TS-mining (Baseline)	20.76	34.57	0.67	34.38	68.67	85.66	35.10	1.12	28.76	75.43	81.77
Pivot (Baseline)	27.74	29.12	4.87	49.47	33.02	71.05	–	–	–	–	–
Si-simp	28.12	**38.20**	4.21	**38.75**	71.63	81.72	38.39	4.16	36.47	74.54	82.90
Trans→Si-simp	29.69	37.63	4.61	35.14	73.15	83.61	39.17	5.25	37.91	74.37	82.94
Para→Si-simp	29.74	37.38	4.22	34.63	73.31	83.58	38.51	4.46	37.39	73.69	82.60
En-simp→Si-simp	29.08	37.31	4.06	35.34	72.52	82.93	38.37	4.79	35.64	**74.70**	**83.26**
Trans→En-simp→Si-simp	29.50	37.81	**4.62**	35.84	72.96	83.32	**39.95**	**5.51**	**38.28**	74.25	82.85
Para→En-simp→Si-simp	30.41	37.33	4.09	33.75	**74.14**	**84.10**	38.63	4.71	36.74	74.46	83.05
Trans→En-simp→Para→Si-simp	28.91	37.65	4.53	36.25	72.18	82.94	39.17	5.37	38.01	74.12	82.93

Si-simp - Sinhala Simplification, *En-simp* - English Simplification, *Trans* - Translation, *Para* - Paraphrasing, BERT - BERTScore. 'T1→T2' indicates that T1, T2 tasks are trained sequentially.

Table 4. Translation Ablation Result

Experiments	mBART					mT5				
	SARI	Components of SARI			BERT	SARI	Components of SARI			BERT
		F_{ADD}	F_{DEL}	F_{KEPT}			F_{ADD}	F_{DEL}	F_{KEPT}	
Trans→Si-simp	37.63	4.61	35.14	73.15	83.61	39.17	5.25	37.91	74.37	82.94
Ta-Si translation	38.24	5.23	37.98	71.51	82.11	39.42	5.99	37.72	74.54	83.24
7k dataset	37.38	4.22	34.63	73.31	83.58	38.51	4.46	37.39	73.69	82.60

Translation Ablation Study: Using translation data provided one of the best performances for sequential training. We explored how the selection of translation data plays a role in performance. For this, we used *Trans→Si-simp* in two different settings: using a different source language for translation and different data sizes for translation. For a different source language for translation, we trained with the Tamil (Ta)-Sinhala data from the same multi-way corpus [26]. For a different data size, we only used 7k En-Si parallel data for the translation task. This dataset size is comparable to the number of paraphrases we used for the model of Martin et al. [47]. The results in Table 4 show that the source language and the size of the dataset used in the translation task have a minimal impact on the TS result.

6.2 Human Analysis

6.2.1 Human Evaluation. We start by looking at the Adequacy, Fluency, and Simplicity of the simplified sentences. In Table 5, we present the average of the human scores. We observe that the human-curated dataset has the highest values in adequacy, fluency, and the average of the three scores. It is also the second best for simplicity, lagging behind TS-mining. However, TS-mining achieves a higher result by simply truncating longer sentences and thus cannot be considered a proper simplification. However, the low value for simplicity relative to that of adequacy and fluency in the human-curated dataset is worth investigating. As mentioned in Section 3, human annotators have mainly focused on sentence splitting when simplifying. However, looking at the complex sentences, most of the words that appear as complex terms are domain-specific terminologies for which a simplified Sinhala term may not exist. This effect is there for model outputs as well–the simplicity score is always lower than the other two scores.

Table 5. Human Evaluation Results of the Models and the Human-curated Dataset

Model	Adequacy	Fluency	Simplicity	Average
sqPLMs (Baseline)	4.55	3.89	1.63	3.36
TS-Mining (Baseline)	3.30	3.54	**3.21**	3.35
Pivot (Baseline)	3.51	3.55	2.27	3.11
Si-simp	4.30	4.37	2.28	3.65
Trans→Si-simp	4.59	4.58	1.99	3.72
Para→Si-simp	4.38	4.34	2.05	3.59
En-simp→Si-simp	4.49	4.35	1.99	3.61
Trans→En-simp→Si-simp	*4.61*	*4.64*	1.99	3.75
Para→En-simp→Si-simp	4.58	4.47	2.05	3.70
Trans→En-simp→Para→Si-simp	4.48	4.54	2.27	*3.76*
Human Simplification	**4.93**	**4.7**	*2.89*	**4.17**

Table 6. Results of the Error Analysis Done by Humans for the Models and Human-curated Dataset

Model	Hallucinations	Fluency Errors	Anaphora Resolution	Bad Substitution	Near-exact Copy	Missing Major Fact
sqPLMs (Baseline)	4	7.34	11.34	13.34	56.66	15.34
TS-Mining (Baseline)	1.33	15.34	8.66	15.34	11.34	74.66
Pivot (Baseline)	24	27.34	4.66	34.66	8	38
Si-simp	7.33	2	4.66	2	54.66	36
Trans→Si-simp	0	3.34	1.34	0	69.34	27.34
Para→Si-simp	0	3.34	1.34	1.34	63.34	28
En-simp→Si-simp	0	5.34	2.66	2	59.34	24.66
Trans→En-simp→Si-simp	0	2	0	0.66	70.66	26.66
Para→En-simp→Si-simp	0	5.34	2	2	64.66	24.66
Trans→En-simp→Para→Si-simp	0.67	2	2.66	0	62	28.66
Human Simplification	4	8	2	9.34	71.34	1.34

Further, we find that all our sequential training models are better than the baseline models. The best average score is when all the auxiliary tasks are sequenced. However, this gain is insignificant when considering sequential fine-tuning with just one intermediate task. Interestingly, the other top-most results are reported by the ITTL models that have the translation task as the first task. These observations tally with the SARI scores reported above.

Consistent with the previous research [36], we also observed that the models tend to copy a large portion of the source sentence while generating the output; i.e., the lesser the number of errors, the higher the amount of copying. We leave further investigations to future research.

6.2.2 Error Analysis by Humans. To further deep dive into the model's output, we performed an error analysis based on the six criteria defined in Section 5.1. In Table 6 and Figure 3, we present the average numbers of sentences per error category type. First and foremost, we see the utility of adding two new criteria ("Near-exact copy" and "Missing major fact") to the error analysis. Models with low scores for hallucination, fluency, anaphora resolution, and bad substitution tend to have a near-exact copy of the complex sentence.

Our error analysis also confirms that our models are better than the baseline models for most cases. In other words, all the baseline models fall into the three worst-performing models with respect to different error categories (*pivot*–five error categories, *prim* and *TS-Mining*–four categories).

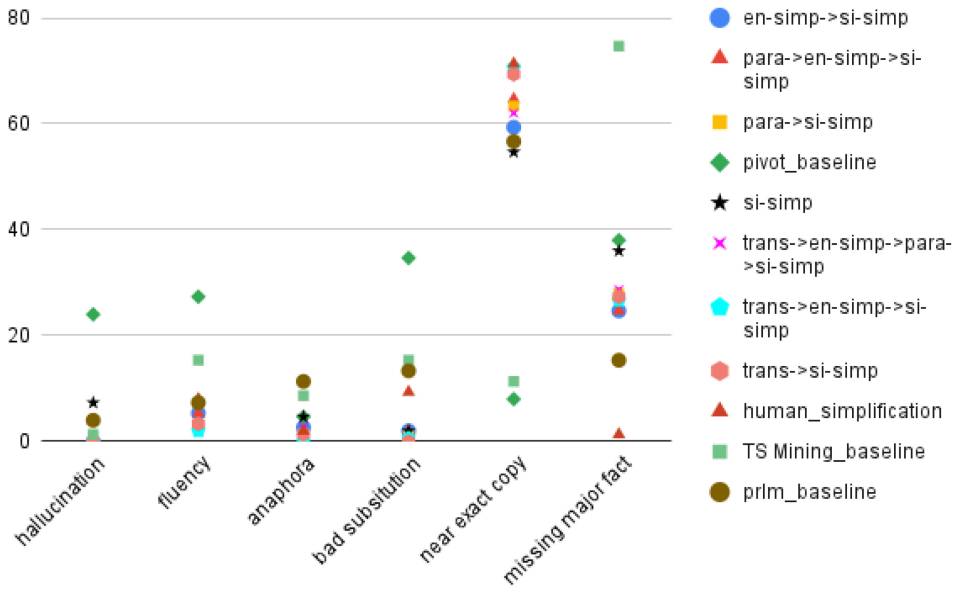

Fig. 3. Visual representation of the number of occurrences of each error type that exist in model outputs and human-curated data.

Complex: මෙම ව්‍යාපෘතිය ක්‍රියාත්මක කිරීම් අරමුණ වන්නේ සමස්ථානික යොදාගෙන , ඊසාන දිග මෝසම් වැස්සන් භූගත ජලයට වන බලපෑම් අධ්‍යයනය යි . (The objective of this project implementation is to study the impact of northeast monsoon rains on groundwater using isotopes.)

TS-Mining (Baseline): ඊසාන දිග මෝසම් වැස්සන් භූගත ජලයට වන බලපෑම් අධ් යයනය (Study the impact of northeast monsoon rains on groundwater)

Si-simp: මෙම ව්‍යාපෘතිය ක්‍රියාත්මක කිරීම් අරමුණ වන්නේ ඊසාන දිග මෝසම් වැස්සන් භූගත ජලයට වන බලපෑම් අධ්‍යයනය කිරීමයි. (The objective of this project implementation is to study the impact of northeast monsoon rains on groundwater.)

Trans→En-simp→Para→Si-simp: ඊසාන දිග මෝසම් වැස්සන් භූගත ජලයට වන බලපෑම් අධ්‍යයනය කිරීම මෙම ව්‍යාපෘතියේ අරමුණයි. (Studying the impact of northeast monsoon rains on groundwater is the objective of this project.)

Fig. 4. Sample sentences generated from three models.

Finally, our comprehensive human evaluation highlights the difficulty of evaluating text simplification systems. The quality of the output of a text simplification system can be analyzed in multiple ways–a model that scores well with respect to one criterion may perform poorly on another criterion. As a consequence, it is difficult to select a single ITTL model as the clear winner of our Sinhala TS experiments. Thus, we agree with Alva-Manchego et al. [5] on the need for better metrics for evaluating text simplification systems. However, overall, using ITTL on sqPLMs is a better solution for text simplification than paraphrasing.

6.3 Qualitative Results

Figure 4 shows the sample output generated for the *TS-Mining*, *Si-Simp*, and *Trans→Si-Simp* models. English translation of each of the sentences is also provided. The sentence generated from

the *TS-Mining* baseline is the shortest, and it misses two major points (use of isotopes and the fact that it is a project). *Si-Simp* and *Trans→Si-Simp* both have one fact missing. However, the *Trans→Si-Simp* model has restructured the sentence in such a way that it appears simpler than the original.

7 Conclusion

TS is largely ignored for low-resource languages due to the unavailability of datasets. In response, we created the SiTSE dataset for Sinhala, a language from the Global South known to be low-resource. Our complex dataset is from government documents, and three human-curated simplified sentences accompany each complex sentence. We modeled the simplification as a zero-resource problem and tested multiple baseline models using mBART and mT5. Further, we improved the performance using ITTL with the use of different auxiliary tasks. We showed that ITTL outperforms the previous work that used sqPLMs for text simplification. Further, we did a human evaluation and added two new categories for the error evaluation to enrich the existing ones. Our automatic and human evaluations show that we should not rely on only one metric to judge the performance of a model. The ITTL method used in this research can be experimented with for other low-resource languages that are included in sqPLMs such as mBART and mT5, as long as there is a small translation dataset.

A limitation of this work is the small size of the dataset—since the dataset has only 1,000 sentences, we had to use it all for testing, thus leaving nothing for model training. As a solution, we are currently working on creating a much larger TS dataset for Sinhala and some other low-resource languages. This will enable us to experiment with the impact of ITTL by considering TS data from different languages. Moreover, this will enable us to experiment with multi-task learning on sqPLMs for text simplification. With some data for model training, it is also possible to experiment with LLMs such as Llama.

References

[1] Sweta Agrawal and Marine Carpuat. 2019. Controlling text complexity in neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. 1549–1564.

[2] Fernando Alva-Manchego, Joachim Bingel, Gustavo Paetzold, Carolina Scarton, and Lucia Specia. 2017. Learning how to simplify from explicit labeling of complex-simplified text pairs. In *Proceedings of the 8th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 295–305.

[3] Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020. 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, 4668–4679. https://doi.org/10.18653/v1/2020.acl-main.424

[4] Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2020. Data-driven sentence simplification: Survey and benchmark. *Computational Linguistics* 46, 1 (2020), 135–187. https://doi.org/10.1162/coli_a_00370

[5] Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2021. The (un)suitability of automatic evaluation metrics for text simplification. *Computational Linguistics* 47, 4 (Dec. 2021), 861–889. https://doi.org/10.1162/coli_a_00418

[6] Yusra Anees and Sadaf Abdul Rauf. 2021. Automatic sentence simplification in low resource settings for Urdu. In *Proceedings of the 1st Workshop on NLP for Positive Impact*. 60–70.

[7] Alessio Palmero Aprosio, Sara Tonelli, Marco Turchi, Matteo Negri, and Mattia A. Di Gangi. 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*. 37–44.

[8] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2020. Translation artifacts in cross-lingual transfer learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP'20)*. 7674–7684.

[9] Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics* 7 (2019), 597–610.

[10] Seyed Ali Bahrainian, Jonathan Dou, and Carsten Eickhoff. 2024. Text simplification via adaptive teaching. In *Findings of the Association for Computational Linguistics (ACL'24)*. 6574–6584.

[11] Alessia Battisti, Dominik Pfütze, Andreas Säuberli, Marek Kostrzewa, and Sarah Ebling. 2020. A corpus for automatic readability assessment and text simplification of german. In *Proceedings of the 12th Language Resources and Evaluation Conference*. 3302–3311.

[12] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research* 3, Feb (2003), 1137–1155.

[13] Stefan Bott and Horacio Saggion. 2011. An unsupervised alignment algorithm for text simplification corpus construction. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*. 20–26.

[14] Laetitia Brouwers, Delphine Bernhard, Anne-Laure Ligozat, and Thomas François. 2014. Syntactic sentence simplification for French. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR'14)*. 47–56.

[15] Dominique Brunato, Andrea Cimino, Felice Dell'Orletta, and Giulia Venturi. 2016. PaCCSS-IT: A parallel corpus of complex-simple sentences for automatic text simplification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 351–361.

[16] Dominique Brunato, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2015. Design and annotation of the first Italian corpus for text simplification. In *Proceedings of the 9th Linguistic Annotation Workshop*. 31–41.

[17] Helena M. Caseli, Tiago F. Pereira, Lucia Specia, Thiago A. S. Pardo, Caroline Gasperin, and Sandra Maria Aluísio. 2009. Building a Brazilian Portuguese parallel corpus of original and simplified texts. *Advances in Computational Linguistics, Research in Computer Science* 41 (2009), 59–70.

[18] Liam Cripwell, Joël Legrand, and Claire Gardent. 2023. Simplicity level estimate (SLE): A learned reference-less metric for sentence simplification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 12053–12059.

[19] Scott A. Crossley, Max M. Louwerse, Philip M. McCarthy, and Danielle S. McNamara. 2007. A linguistic analysis of simplified and authentic texts. *Modern Language Journal* 91, 1 (2007), 15–30.

[20] Nisansa de Silva. 2021. Survey on publicly available Sinhala Natural Language Processing tools and research. *arXiv preprint arXiv:1906.02358v10* (2021).

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long and Short Papers)*. Association for Computational Linguistics, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[22] Vinura Dhananjaya, Piyumal Demotte, Surangika Ranathunga, and Sanath Jayasena. 2022. BERTifying Sinhala–A comprehensive analysis of pre-trained language models for Sinhala text classification. In *Proceedings of the 13th Language Resources and Evaluation Conference*. 7377–7385.

[23] Vinura Dhananjaya, Surangika Ranathunga, and Sanath Jayasena. 2024. Lexicon-based fine-tuning of multilingual language models for low-resource language sentiment analysis. *CAAI Transactions on Intelligence Technology* 9, 5 (2024), 1116–1125.

[24] Yue Dong, Zichao Li, Mehdi Rezagholizadeh, and Jackie Chi Kit Cheung. 2019. 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*. 3393–3402.

[25] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).

[26] Aloka Fernando, Surangika Ranathunga, and Gihan Dias. 2020. Data augmentation and terminology integration for domain-specific Sinhala-English-Tamil statistical machine translation. *arXiv preprint arXiv:2011.02821* (2020).

[27] Núria Gala, Anaïs Tack, Ludivine Javourey-Drevet, Thomas François, and Johannes C. Ziegler. 2020. Alector: A parallel corpus of simplified French texts with alignments of misreadings by poor and dyslexic readers. In *Language Resources and Evaluation for Language Technologies (LREC'20)*.

[28] Cristina Garbacea, Mengtian Guo, Samuel Carton, and Qiaozhu Mei. 2021. Explainable prediction of text complexity: The missing preliminaries for text simplification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 1086–1097.

[29] Isao Goto, Hideki Tanaka, and Tadashi Kumano. 2015. Japanese news simplification: Tak design, data set construction, and analysis of simplified text. In *Proceedings of Machine Translation Summit XV: Papers*.

[30] Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018. Dynamic multi-level multi-task learning for sentence simplification. In *Proceedings of the 27th International Conference on Computational Linguistics*. 462–476.

[31] Sebastian Joseph, Kathryn Kazanas, Keziah Reina, Vishnesh Ramanathan, Wei Xu, Byron C. Wallace, and Junyi Jessy Li. 2023. Multilingual simplification of medical texts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 16662–16692.

[32] David Kauchak. 2013. Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1537–1546.
[33] Tannon Kew, Alison Chi, Laura Vásquez-Rodríguez, Sweta Agrawal, Dennis Aumiller, Fernando Alva-Manchego, and Matthew Shardlow. 2023. BLESS: Benchmarking large language models on sentence simplification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 13291–13309.
[34] David Klaper, Sarah Ebling, and Martin Volk. 2013. Building a German/Simple German parallel corpus for automatic text simplification. *ACL 2013* (2013), 11.
[35] Sigrid Klerke and Anders Søgaard. 2012. DSim, a Danish parallel corpus for text simplification. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)*. 4015–4018.
[36] Reno Kriz, João Sedoc, Marianna Apidianaki, Carolina Zheng, Gaurav Kumar, Eleni Miltsakaki, and Chris Callison-Burch. 2019. Complexity-weighted loss and diverse reranking for sentence simplification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long and Short Papers)*. 3137–3147.
[37] Jeongwon Kwak, Hyeryun Park, Kyungmo Kim, and Jinwook Choi. 2023. Context and literacy aware learnable metric for text simplification. In *Proceedings of the 3rd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM'23)*. 175–180.
[38] En-Shiun Lee, Sarubi Thillainathan, Shravan Nayak, Surangika Ranathunga, David Adelani, Ruisi Su, and Arya D. McCarthy. 2022. Pre-trained multilingual sequence-to-sequence models: A hope for low-resource language translation? In *Findings of the Association for Computational Linguistics (ACL'22)*. 58–67.
[39] M. Lewis. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461* (2019).
[40] Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).
[41] Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics* 8 (Nov. 2020), 726–742. https://doi.org/10.1162/tacl_a_00343.
[42] Mounica Maddela, Fernando Alva-Manchego, and Wei Xu. 2021. Controllable text simplification with explicit paraphrasing. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 3536–3553. https://doi.org/10.18653/v1/2021.naacl-main.277
[43] Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 16383–16408.
[44] Jonathan Mallinson and Mirella Lapata. 2019. Controllable sentence simplification: Employing syntactic and lexical constraints. *arXiv preprint arXiv:1910.04387* (2019).
[45] Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. 2020. Zero-shot crosslingual sentence simplification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP'20)*. 5109–5126.
[46] Louis Martin, Éric de la Clergerie, Benoît Sagot, and Antoine Bordes. 2020. Controllable sentence simplification. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association, 4689–4698. https://aclanthology.org/2020.lrec-1.577
[47] Louis Martin, Angela Fan, Éric Villemonte De La Clergerie, Antoine Bordes, and Benoît Sagot. 2022. MUSS: Multilingual unsupervised sentence simplification by mining paraphrases. In *Proceedings of the 13th Language Resources and Evaluation Conference*. 1651–1664.
[48] Jana M. Mason and Janet R. Kendall. 1978. *Facilitating Reading Comprehension through Text Structure Manipulation*. Technical Report No. 92. (1978).
[49] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems* 26 (2013).
[50] Martina Miliani, Serena Auriemma, Fernando Alva-Manchego, and Alessandro Lenci. 2022. Neural readability pairwise ranking for sentences in Italian administrative language. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*. 849–866.
[51] Sneha Mondal, Ritika Ritika, Ashish Agrawal, Preethi Jyothi, and Aravindan Raghuveer. 2024. DIMSIM: Distilled multilingual critics for Indic text simplification. In *Findings of the Association for Computational Linguistics (ACL'24)*. 16093–16109.
[52] Akifumi Nakamachi, Tomoyuki Kajiwara, and Yuki Arase. 2020. Text simplification with reinforcement learning using supervised rewards on grammaticality, meaning preservation, and simplicity. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: Student Research Workshop*. 153–159.

[53] Shashi Narayan and Claire Gardent. 2016. Unsupervised sentence simplification using deep semantics. In *Proceedings of the 9th International Natural Language Generation Conference*. 111–120.

[54] Shravan Nayak, Surangika Ranathunga, Sarubi Thillainathan, Rikki Hung, Anthony Rinaldi, Yining Wang, Jonah Mackey, Andrew Ho, and En-Shiun Annie Lee. 2023. Leveraging auxiliary domain parallel data in intermediate task fine-tuning for low-resource translation. *arXiv preprint arXiv:2306.01382* (2023).

[55] Daiki Nishihara, Tomoyuki Kajiwara, and Yuki Arase. 2019. Controllable text simplification with lexical constraint loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*. 260–266.

[56] Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P. Dinu. 2017. Exploring neural text simplification models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 85–91.

[57] Kashyapa Niyarepola, Dineth Athapaththu, Savindu Ekanayake, and Surangika Ranathunga. 2022. Math word problem generation with multilingual language models. In *Proceedings of the 15th International Conference on Natural Language Generation*. 144–155.

[58] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. Fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Association for Computational Linguistics, 48–53. https://doi.org/10.18653/v1/N19-4009

[59] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. 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, 311–318. https://doi.org/10.3115/1073083.1073135

[60] Jason Phang, Iacer Calixto, Phu Mon Htut, Yada Pruksachatkun, Haokun Liu, Clara Vania, Katharina Kann, and Samuel Bowman. 2020. English intermediate-task training improves zero-shot cross-lingual transfer too. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. 557–575.

[61] Jason Phang, Thibault Févry, and Samuel R. Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *arXiv preprint arXiv:1811.01088* (2018).

[62] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog* 1, 8 (2019), 9.

[63] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21, 140 (2020), 1–67.

[64] Surangika Ranathunga and Nisansa De Silva. 2022. Some languages are more equal than others: Probing deeper into the linguistic disparity in the NLP world. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 823–848.

[65] Surangika Ranathunga, Nisansa De Silva, Velayuthan Menan, Aloka Fernando, and Charitha Rathnayake. 2024. Quality does matter: A detailed look at the quality and utility of web-mined parallel corpora. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*. 860–880.

[66] Luz Rello, Ricardo Baeza-Yates, Laura Dempere-Marco, and Horacio Saggion. 2013. Frequent words improve readability and short words improve understandability for people with dyslexia. In *IFIP Conference on Human-computer Interaction*. Springer, 203–219.

[67] Michael Ryan, Tarek Naous, and Wei Xu. 2023. Revisiting non-English text simplification: A unified multilingual benchmark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 4898–4927.

[68] Andreas Säuberli, Sarah Ebling, and Martin Volk. 2020. Benchmarking data-driven automatic text simplification for German. In *Proceedings of the 1st Workshop on Tools and Resources to Empower People with REAding DIfficulties (READI'20)*. 41–48.

[69] Carolina Scarton, Gustavo Paetzold, and Lucia Specia. 2018. Simpa: A sentence-level simplification corpus for the public administration domain. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC'18)*.

[70] Carolina Scarton and Lucia Specia. 2018. Learning simplifications for specific target audiences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: short Papers)*. 712–718.

[71] Lane Schwartz, Emily Chen, Benjamin Hunt, and Sylvia L. R. Schreiner. 2019. Bootstrapping a neural morphological analyzer for St. Lawrence Island Yupik from a finite-state transducer. In *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages (Volume 1: Papers)*. Association for Computational Linguistics, 87–96. https://aclanthology.org/W19-6012

[72] Advaith Siddharthan, Ani Nenkova, and Kathleen McKeown. 2004. Syntactic simplification for improving content selection in multi-document summarization. In *Proceedings of the 20th International Conference on Computational Linguistics*. 896–es.

[73] Lucia Specia. 2010. Translating from complex to simplified sentences. In *Proceedings of the 9th International Conference on Computational Processing of the Portuguese Language (PROPOR'10)*. Springer, 30–39.

[74] Regina Stodden, Omar Momen, and Laura Kallmeyer. 2023. DEplain: A German parallel corpus with intralingual translations into plain language for sentence and document simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 16441–16463.

[75] Elior Sulem, Omri Abend, and Ari Rappoport. 2018. 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, 738–744. https://doi.org/10.18653/v1/D18-1081

[76] Elior Sulem, Omri Abend, and Ari Rappoport. 2018. Simple and effective text simplification using semantic and neural methods. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 162–173.

[77] Sai Surya, Abhijit Mishra, Anirban Laha, Parag Jain, and Karthik Sankaranarayanan. 2019. Unsupervised neural text simplification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2058–2068.

[78] Sotaro Takeshita, Tommaso Green, Niklas Friedrich, Kai Eckert, and Simone Paolo Ponzetto. 2022. X-SCITLDR: Cross-lingual extreme summarization of scholarly documents. In *Proceedings of the 22nd ACM/IEEE Joint Conference on Digital Libraries*. 1–12.

[79] Sarubi Thillainathan, Surangika Ranathunga, and Sanath Jayasena. 2021. Fine-tuning self-supervised multilingual sequence-to-sequence models for extremely low-resource NMT. In *2021 Moratuwa Engineering Research Conference (MERCon'21)*. IEEE, 432–437.

[80] Sara Tonelli, Alessio Palmero Aprosio, and Francesca Saltori. 2016. SIMPITIKI: a simplification corpus for Italian. In *Proceedings of the Third Italian Conference on Computational Linguistics CLiC-it*. 291–296.

[81] Sowmya Vajjala and Ivana Lučić. 2018. OneStopEnglish corpus: A new corpus for automatic readability assessment and text simplification. In *Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications*. 297–304.

[82] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* 30 (2017).

[83] Tu Vu, Baotian Hu, Tsendsuren Munkhdalai, and Hong Yu. 2018. Sentence simplification with memory-augmented neural networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*. 79–85.

[84] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

[85] Kristian Woodsend and Mirella Lapata. 2011. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. 409–420.

[86] Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics* 3 (2015), 283–297. https://doi.org/10.1162/tacl_a_00139

[87] Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics* 4 (2016), 401–415. https://doi.org/10.1162/tacl_a_00107

[88] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 483–498. https://doi.org/10.18653/v1/2021.naacl-main.41

[89] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SkeHuCVFDr

[90] Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 584–594.

[91] Yaoyuan Zhang, Zhenxu Ye, Yansong Feng, Dongyan Zhao, and Rui Yan. 2017. A constrained sequence-to-sequence neural model for sentence simplification. *arXiv preprint arXiv:1704.02312* (2017).

[92] Sanqiang Zhao, Rui Meng, Daqing He, Andi Saptono, and Bambang Parmanto. 2018. Integrating transformer and paraphrase rules for sentence simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 3164–3173.

[93] Xinran Zhao, Esin Durmus, and Dit-Yan Yeung. 2023. Towards reference-free text simplification evaluation with a BERT Siamese network architecture. In *Findings of the Association for Computational Linguistics (ACL'23)*. 13250–13264.

[94] Yanbin Zhao, Lu Chen, Zhi Chen, and Kai Yu. 2020. Semi-supervised text simplification with back-translation and asymmetric denoising autoencoders. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 9668–9675.

[95] Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*. 1353–1361.

Appendices

A Evaluation Metrics

SARI was introduced by Xu et al. [87]. SARI compares the output of a TS system against the input sentence as well as the references. It measures the goodness of added/deleted/kept words during text simplification. In SARI, words that are in the output but not in the input sentence are rewarded, if these words occur in the references. The corresponding value F_{ADD} is calculated by taking the n-gram precision and recall of addition operations. Similarly, words that are in both the input and output are rewarded, if these words occur in the references. The corresponding value F_{KEPT} is calculated by taking the precision and recall of the retained n-grams. Deleted words in the output are rewarded if these words are missing in the reference as well. However, F_{DEL} considers only precision, because over-deleting hurts readability much more significantly than not deleting. SARI is the arithmetic average of these values.

BERTScore is a reference-based metric that computes the cosine similarity between tokens in a system output and in a manual reference using contextual embeddings [5]. A higher BERTScore means good output sentence quality.

B Error Categories

The error categories used in this research are listed below

— *Fluency Errors:* The output contains grammatical errors. Repetitions too are included in this category.
— *Hallucinations:* The output contains information that was not included in the source sentence. This measures whether the model generates out-of-context words or phrases.
— *Anaphora Resolution:* The output contains pronouns that are hard to resolve.
— *Bad Substitution:* The output has substituted a phrase that is not simpler than the input.
— *Near-exact Copy:* The output is the same as the complex sentence, with few common words (e.g., stop words) changed.
— *Missing Major Fact:* The output misses a fact that was in the complex sentence, which is important to meaning preservation.

C Human Participant Details

Creating the Simplified Dataset: All annotators were native Sinhala speakers and had a (minimum) undergraduate degree.

First Error Analysis: We employed three evaluators for this task, not used in the dataset creation task. Two of them have recently completed their undergraduate degree (IT and Architecture), and the third was a final-year undergraduate (Engineering), and their primary mode of university education is English.

Second Error Analysis: We employed three evaluators not used in the above two tasks. One was a Sinhala lecturer, the other was a Sinhala undergraduate, and the third was one of the authors.

All but one human participant (who preferred co-authorship) were compensated adhering to the institution policies.

D Implementation Details

Hyper-parameters: For fine-tuning the mBART model we used the adam-optimizer, 3e-05 as the initial learning rate, 0.3 as dropout, 2,500 as warmup updates, inverse square root as the learning rate scheduler algorithm, and 32 as the batch size. For the mT5 model we used Adafactor as the optimizer, 3e-05 as the initial learning rate, 0.1 as dropout, 2,500 as warmup updates, linear as the learning rate scheduler algorithm, and 4 as the batch size. Due to resource limitations, we used half precision to fine-tune the mBART model. mT5 experiments were able to run using full precision.

Computational Resources Used: We used AWS *g4dn.2xlarge* instances to carry out the experiments. We estimate that 340 GPU hours on an NVIDIA Tesla T4 were spent on this research for successful fine-tuning tasks, paraphrase mining, and erroneous attempts. For paraphrase mining, we used a single GPU instance–an NVIDIA Tesla T4 with 32 vCPUs and 120 GB RAM.

Received 21 November 2023; revised 7 November 2024; accepted 2 March 2025