Revision for Concision: A Constrained Paraphrase Generation Task

Wenchuan Mu   Kwan Hui Lim
Singapore University of Technology and Design
{wenchuan_mu,kwanhui_lim}@sutd.edu.sg

Abstract

Academic writing should be concise as concise sentences better keep the readers' attention and convey meaning clearly. Writing concisely is challenging, for writers often struggle to revise their drafts. We introduce and formulate revising for concision as a natural language processing task at the sentence level. Revising for concision requires algorithms to use only necessary words to rewrite a sentence while preserving its meaning. The revised sentence should be evaluated according to its word choice, sentence structure, and organization. The revised sentence also needs to fulfil semantic retention and syntactic soundness. To aide these efforts, we curate and make available a benchmark parallel dataset that can depict revising for concision. The dataset contains 536 pairs of sentences before and after revising, and all pairs are collected from college writing centres. We also present and evaluate the approaches to this problem, which may assist researchers in this area.

1 Introduction

Concision and clarity[1] are important in academic writing as wordy sentences will obscure good ideas (Figure 1). Concise writing encourages writers to choose words deliberately and precisely, construct sentences carefully to eliminate deadword, and use grammar properly (Stanford University), which often requires experience and time. A first draft often contains far more words than necessary, and achieving concise writing requires revisions (MON, 2020). As far as we know, currently this revision process can only be done manually, or semi-manually with the help of some rule-based wordiness detectors (Adam and Long, 2013). We therefore introduce and formulate revising for concision as a natural language processing (NLP) task
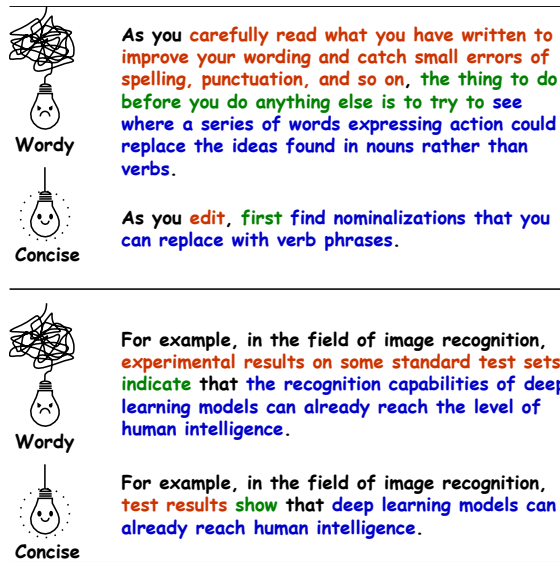

Figure 1: Wordy sentences are more boring to read than concise sentences. But how do we turn lengthy sentences into concise ones? We show two examples. The above sentence pair is taken from the Purdue Writing Lab, which suggests how college students should succinctly revise their writing (PU). In the other example, the wordy sentence comes from a scientific paper (Chen et al., 2020), and its concise counterpart is predicted from the concise revisioner we developed (Section 5). In each pair, text with the same colour delivers the same information.

and address it. In this study, we make the following contributions:

1. We formulate the revising for concision NLP task at the sentence level, which reflects the revising task in academic writing. We also survey the differences between this task and sentence compression, paraphrasing, etc.

2. We release a corpus of 536 sentence pairs, curated from 72 writing centres and additionally coded with the various linguistic rules for concise sentence revision.

3. We propose an gloss-based Seq2Seq approach

---

[1]We treat *concision* and *conciseness* as equivalent, and *clarity* as part of *concision*

to this problem, and conduct automatic and human evaluations. We observed promising preliminary results and we believe that our findings will be useful for researchers working in this area.

2 Problem Statement

2.1 Revision as an English Writing Task

Concise writing itself is a lesson that is often emphasized in colleges, and revision is crucial in writing. The following definitions are helpful when we set out to formulate the task.

Definition 2.1 (Concise). Marked by brevity of expression or statement: free from all elaboration and superfluous detail (Merriam-Webster).

Definition 2.2 (Concise writing, English). Writing that is clear and does not include unnecessary or vague/unclear words or language (UOA).

Revising for concision at paragraph level, or even article level, may be the best practice. However, sentence-level revising usually suffices. We focus on revising for concision at the sentence level now. Indeed, in many college academic writing tutorials, revisions for concision are for individual sentences, and this process is defined as follows.

Definition 2.3 (Revise for concision at the sentence level, English[2]). Study a sentence in draft, use specific strategies[3] to edit the sentence concisely without losing meaning.

If someone, such as a college student, wants to concisely modify a sentence, specific strategies (*e.g.*, *delete* weak modifiers, *replace* phrasal verbs with single verbs, or *rewrite* in active voice, etc.) tell us how to locate wordiness and how to edit it (PU; WU; UALR; UNZ; MON, 2020). The rule is to repeatedly detect wordiness and revise it until no wordiness is detected or it cannot be removed without adding new wordiness. The final product serves as a concise version of the original sentence, if it does not lose its meaning.

2.2 Task Definition in NLP

Now that we know how humans can revise a sentence, what about programs? Each strategy is clear to a trained college student, but not clear enough to program in code. On the one hand, existing verbosity detectors may suggest which part of a sentence is too "dense" (Adam and Long, 2013), but

fail to expose fine-grained wordiness details. On the other hand, how programs can edit sentences without losing their meaning remains challenging. In short, no existing program can generate well-modified sentences in terms of concision.

Eager for a program that revises sentences nicely and concisely, we set out to formulate this modification process as a sequence-to-sequence (Seq2Seq) NLP task. In this task, the input is any English sentence and the output should be its concise version. We define it as follows.

Definition 2.4 (Revise for concision at the sentence level, NLP). Produce a sentence where minimum wordiness can be identified. (And,) the produced sentence delivers the same information as input does. (And,) the produced sentence is syntactically correct.

As many other NLP tasks, *e.g.*, machine translation, named-entity recognition, etc., Definition 2.4 describes the product (text) of a process, not the process itself, *i.e.*, how the text is produced. This perspective is different from that of Definition 2.3.

Among the three components in Definition 2.4, both the first and the third are clear and self-contained. They are related to syntax; hence, at least human experts would think it straightforward to determine the soundness of a sentence on both. For example, the syntax correctness of an English sentence will not be judged differently by different experts, unless the syntax itself changes. Unfortunately, the second component is neither clear nor self-contained. This component asks for information retention, which is a rule inherited from Definition 2.3. Determining the semantic similarity between texts has long been challenging, even for human experts (Rus et al., 2014).

We then clarify the definition by assuming that combining the second and third components in Definition 2.4 meet the definition of the paraphrase generation task (Rus et al., 2014). Henceforth, Definition 2.4 can be simplified to Definition 2.5.

Definition 2.5 (Revise for concision at the sentence level, NLP, simplified). Produce a *paraphrase* where minimum wordiness can be identified.

The revising[4] task is well-defined, as long as "paraphrase generation" is well-defined. It is a paraphrase generation task with a syntactic constraint.

---

[2] Adapted from notes of PU Writing Lab and Rambo (2019)

[3] Presented in Appendix (Table 4) as a periphery of this study.

[4] stands for *(machine) revising for concision* if not otherwise specified, so does *revision*

2.3 Task Performance Indicator

How does one approximately measure revision performance? In principle, Definition 2.4 should be used as a checklist. A good sample requires correct grammar (γ), complete information (ρ) and reduced wordiness ($1 - \omega$), assuming each component as a float number between 0 and 1. The overall assessment (χ) of the three components is as follows,

$$\chi = \alpha^2 \cdot (\gamma - 1) + \alpha \cdot (\rho - 1) + (1 - \omega), \quad (1)$$

where $\alpha \in \mathbb{R}_{>1}$ is a large enough number, as we believe that γ and ρ overweigh $1 - \omega$. Intuitively, if a revised sentence does not paraphrase the original one, assessing the reduction of wordiness makes little sense. Concision χ would always be negative if $\gamma < 1$ or $\rho < 1$.

Corresponding to the three components is a mix of three tasks, including grammatical error correction for g, textual semantic similarity for r, and wordiness detection for w. Unfortunately, both a reference-free metric good enough to characterize the paraphrase and a robust wordiness detector are rare. Therefore, such assessment of concision is now only feasible through human evaluation.

To enable automatic evaluation for faster feedback, we currently follow Papineni's viewpoint (Papineni et al., 2002). The closer a machine revision is to a professional human revision, the better it is. To judge the quality of a machine revision, one measures its closeness to one or more reference human revisions according to a numerical metric. Thus, our revising evaluation system requires two main components:

1. A numerical "revision closeness" metric.

2. A corpus of good quality human reference revisions.

Different from days when Papineni needed to propose a closeness metric, we can adopt various metrics from machine translation and summarization community (Lin, 2004; Banerjee and Lavie, 2005). Since it is certain which criterion correlates best, we take multiple relevant and reasonable metrics into account to estimate quality of revision. These metrics include those measuring higher order n-grams precision (BLEU, Papineni et al., 2002), explicit Word-matching, stem-matching, or synonym-matching (METEOR, Banerjee and Lavie, 2005), surface bigram units overlapping

(ROUGE-2-F1, Lin, 2004), cosine similarity between matched contextual words embeddings (BERTScore-F1, Zhang et al., 2020b), edit distance with single-word insertion, deletion, or replacement (word error rate, Su et al., 1992), edit distance with block insertion, deletion, or replacement (translation edit rate, Snover et al., 2006), and explicit goodness of words editing against reference and source (SARI, Xu et al., 2016). In short, BLEU, METEOR, ROUGE-2-F1, SARI, word error rate and translation edit rate estimate sentence well-formedness lexically; METEOR and BERTScore-F1 consider semantic equivalence. Comparing grammatical relations found in prediction with those found in references can also measure semantic similarity (Clarke and Lapata, 2006b; Riezler et al., 2003; Toutanova et al., 2016). Grammatical relations are extracted from dependency parsing, and F1 scores can then be used to measure overlap.

In contrast, the lack of good parallel corpus impedes (machine) revising for concision. To address this limitation, we curate and make available such a corpus as benchmark. Each sample in the corpus contains a wordy sentence, and at least one sentence revised for concision. Samples are from English writing centres of 57 universities, ten colleges, four community colleges, and a postgraduate school.

3 Related Work

Manual revision operations include delete, replace, and rewrite. Intuitively, a revising program should do similar jobs, too. In fact, these actions are implemented individually in various NLP tasks. For example, sentence compression requires programs to delete unnecessary words, and paraphrasing itself is a matter of replacement. Machine revision for concision could also share traits with them. Practically, when a neural model learns in a Seq2Seq manner, the difference among these tasks is the parallel dataset. We are also interested in whether programs developed for these tasks can work in machine revision.

3.1 Deleting as in Sentence Compression

When revising, deleting redundant words is common. For example, we can revise "*research is increasing in the field of nutrition and food science*" to "*research is increasing in nutrition and food science*" (URI, 2019), simply by deleting "*the field of*". Deleting is canonical in sentence com-

pression, a task aiming to reduce sentence length from source sentences while retaining basic meaning (Jing, 2000; Knight and Marcu, 2000; McDonald, 2006). For example, the compression task has been formulated as integer linear programming optimization using syntactic trees (Clarke and Lapata, 2006a), or as a sequence labelling optimization problem using the recurrent neural networks (RNN) (Filippova et al., 2015; Klerke et al., 2016; Kamigaito et al., 2018). They explicitly or implicitly use dependency grammar. Pre-trained language models such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) can encode features apart from dependency parsing (Kamigaito and Okumura, 2020), bringing prediction and reference sentences closer.

All methods rely on parallel datasets labelling parts to be deleted. However, the deleting part in sentence compression differs from that in revision. Filippova and Altun (2013) created Google dataset from titles and first sentence of news articles. The information retained in the first sentence depends on the title. While this creation is useful for reducing excessive information, the deleted part is probably not wordiness.

Deleting does not solve everything in revision. We can revise "*in this report I will conduct a study of ants and the setup of their colonies*" to "*in this report I will study ants and their colonies*", taking advantage of noun-and-verb homograph. However, a more concise version "*this report studies ants*" (Commnet) requires changing "*study*" to third-person singular.

3.2 Replacing as in Paraphrase Generation

Word choice matters as well, thus we revise by paraphrasing to stronger words. Paraphrase generation changes a sentence grammatically and reselects words, while retaining meaning. Paraphrasing matters in academic writing, for it helps avoid plagiarism. Rule-based or statistical machine paraphrasing substitutes words by finding synonyms from lexical databases, and decodes syntax according to template sentences. This rigid method may undermine creativity (Bui et al., 2021). Pre-trained neural language models like GPT (Radford et al., 2019) or BART (Lewis et al., 2020) paraphrase more accurately (Hegde and Patil, 2020). Through paraphrasing, we can replace verb phrase "*conduct a study*" to verb "*study*" in the example above, rather than delete and rely on noun-and-verb homo-

graphs to keep the sentence syntactically correct.

Machine revision is a kind of paraphrase generation, and vice versa is not true. Current paraphrase generation does not require concision in generated sentences. Automatically annotated datasets for paraphrasing include ParaNMT (Wieting and Gimpel, 2018), Twitter (Lan et al., 2017), or re-purposed noisy datasets such as MSCOCO (Lin et al., 2014) and WikiAnswers (Fader et al., 2013). We may adapt paraphrase parallel datasets to train revising models, as investigated in Section 5.

3.3 Other related tasks

Summarization produces a shorter text of one or several documents, while retaining most of meaning (Paulus et al., 2018). This is similar to sentence compression. In practice, summarization welcomes novel words, allows specifying output length (Kikuchi et al., 2016), and removes much more information than sentence compression does. Datasets include XSum (Narayan et al., 2018), CNN/DM (Hermann et al., 2015), WikiHow (Koupaee and Wang, 2018), NYT (Sandhaus, 2008), DUC-2004 (Over et al., 2007), and Gigaword (Rush et al., 2015), where summaries are generally shorter than one-tenth of documents. On the other hand, sentence summarization (Chopra et al., 2016) uses summarization methods on sentence compression datasets, retaining more information and possibly generating new words.

Text simplification modifies vocabulary and syntax for easier reading, while retaining approximate meaning (Omelianchuk et al., 2021). Handcrafted syntactic rules (Siddharthan, 2006; Carroll et al., 1999; Chandrasekar et al., 1996) and aligned sentences-driven simplification (Yatskar et al., 2010) have been explored. Corpora such as Turk (Xu et al., 2016) and PWKP (Zhu et al., 2010) are compiled from Wikipedia and Simple English Wikipedia (Coster and Kauchak, 2011). Rules for simplification may deviate from that for revision, *e.g.*, text simplification sometimes encourages prepositional phrases (Xu et al., 2016). Still, adapting these approaches may benefit academic revising for concision.

Fluency editing (Napoles et al., 2017) not only corrects grammatical errors but paraphrases text to be more native sounding as well. Its paraphrasing section is constrained such that outputs represent a higher level of English proficiency than inputs. As a constrained paraphrase task, fluency editing may

alleviate ill-posed problems in paraphrase generation (Cao et al., 2020; Rus et al., 2014). However, such constraints may not be consistent with those required for concision.

In general, machine revision for academic writing requires new methods. Rules for revision can be adapted from these related tasks, so do training strategies.

4 Benchmark Corpus

The collated corpus, named **Concise-536**, contains 536 pairs of sentences. This is a fair starting size, comparable with 385 of RST-DT (semantic parsing, Carlson et al., 2003), 500 of DUC 2004 (summarization[5]), or 575 by Cohn and Lapata (2008) (sentence compression). Each concise sentence is revised from its wordy counterpart by English specialists from the 72 universities, colleges, or community colleges. Sentence ID, category and original link are available for each data point[6], and a 120-point validation split from other sources is attached.

Revising different sentences can go through a completely different process. As seen below, simply crossing out a few words revises Example 4.1; a new word is needed in revising Example 4.2; and even the sentence structure needs changing in Example 4.3.

Corpus Example 4.1 (Delete). Any ~~particular type of~~ dessert is fine with me. (PU)

Corpus Example 4.2 (Replace). She ~~has the ability to~~ can influence the outcome. (PU)

Corpus Example 4.3 (Rewrite). ~~The 1780 constitution of Massachusetts was written by John Adams.~~ John Adams wrote the 1780 Massachusetts Constitution. (UNC, 2021)

In Concise-536, we do not identify fine-grained wordiness because a phrase can have more than one type of verbosity at the same time. For instance, we can revise "*Her poverty also helped in the formation of her character.*" to "*Her poverty also helped form her character.*" (George Mason University, 2021), treating "*in the formation of*" as either a wordy prepositional phrase, or nominalization. Rather, we focus on editing.

In editing, the three actions are not complementary, and instead have varying degrees of power. Deleting can be covered by replacing (See Section 3.1), which could be again covered by rewrit-

ing, *i.e.*, rewriting is the most flexible. However, Occam's razor pushes us to prioritize the actions requiring lower effort to complete, *i.e.*, delete $<$ replace $<$ rewrite. Supposedly, the difficulty for implement each action with programs shares the same trend. In addition, some sentences contain multiple wordiness occurrences, each of which may need a different action, *e.g.*, delete + replace.

Interested in how well a revising algorithm resembles each action, we label revisions in each sentence pair and divide them into seven categories. Revisions that require the same set of actions will be assigned to the same category. Each revision is assigned to one of seven categories in Table 1.

For convenience, we standardize categorizing rules as follows where each sentence is a word-level sequence. For each pair, we have a wordy sequence (w) and a concise sequence (c).

1. If c is a (not necessarily consecutive) subsequence of w, we consider revision only requires deletion (category I).

2. If not, we only delete redundancy from w to get w', *i.e.*, w' paraphrases w, and w' is a subsequence of w. Then, we make local[7] replacement(s) to w' to get w^*, and every individual state from w' to w^* (*i.e.*, after each local replacement) paraphrases w'. If $w^* = c$ and $w' = w$, we consider revision only requires replacement (category II). If $w^* = c$ and $w' \neq w$, we consider revision only requires deletion and replacement (category IV).

3. If $w^* = w$, we consider revision relies solely on rewriting (category III).

Corpus Example 4.4 (category I). ~~There are~~ four rules ~~that~~ should be observed. (PU)

Corpus Example 4.5 (category III). ~~Regular reviews of~~ online content should be ~~scheduled~~ reviewed regularly. (MON, 2020)

Corpus Example 4.6 (category IV). She fell ~~down due to the fact that~~ because she hurried. (PU)

Example 4.4 used to be wordy in the running start, but deleting suffices in revision. Therefore, although counter intuitive, it belongs to category I. An adjective-noun pair is the wordiness in Example 4.5, yet its revision is more complex than replacing a verb. Usually, revision involves multiple

---

[5]https://duc.nist.gov/duc2004/
[6]https://huggingface.co/datasets

[7]Empirically, in a sentence or clause, we do not replace the subject and predicate verb together.

Category	Action	# sents.	Mean words wordy sent.	Mean words concise sent.	Translation Edit Rate
I	Delete	169	13.16	9.17	4.72
II	Replace	116	12.37	9.02	5.1
III	Rewrite	153	14.43	9.73	9.54
IV	Delete + Replace	42	23.81	11.57	15.16
V	Replace + Rewrite	33	21.52	12.85	14.88
VI	Delete + Rewrite	14	24.5	11.36	17.71
VII	Delete + Replace + Rewrite	9	32.56	14.56	25.56
All	-	536	15.32	9.86	8.31

Table 1: Revising a sentence can involve either one of the three strategies (category I, II, III), or a combination of them (category IV, V, VI, VII). Sample sizes, average word counts before and after revisions, and average edit distance (translate edit rate, TER) for revision are listed.

strategies, as seen in Example 4.6 (delete "*down*" + replace "*due to the fact that with*" with "*because*").

Human annotators implement the rules, as we need to check whether the meaning is still the same at each step.

Usually, category III sentences are the hardest to revise as the easier strategies of deleting and replacing are not applicable. In fact, revising category V, VI, and VII sentences are more challenging, as these sentences are longer, more complex, and more deliberate than category III sentences (Figure 4, Table 9, 10), which is a bias in this corpus.

5 Approaches to Revisions

We approach the raised problem in this study. Solutions to machine revision for concision can be diverse. Neural model solutions include tree-to-tree transduction models (Cohn and Lapata, 2008), or general Seq2Seq models. We present a Seq2Seq approach, for it is flexible and straightforward. The model architecture is BART (Lewis et al., 2020).

Ideally, training corpora tune statistical models or neural models, such that we can test tuned models on the benchmark corpus. However, lacking authoritative revisions prompted us to let models fit relevant task data. We also use public external knowledge, *e.g.*, WordNet (Fellbaum, 2010). This section describes how we build an ad hoc training corpus to initiate this task.

The BART base model (124,058,116 parameters) is then used to fit *each* training set in this section. Training settings are fixed (batch size at 32, PyTorch Adam optimizer (Paszke et al., 2019; Kingma and Ba, 2015) with initial learning rate at 5×10^{-5}, validated every 5,000 iterations). We then evaluate trained models on Concise-536.

5.1 Baselines

We prepare training samples by adjusting data from paraphrase generation (ParaNMT, Wieting and Gimpel, 2018), sentence simplification (Wik-iSmall, Zhang and Lapata, 2017), or sentence compression (Gigaword, Rush et al., 2015; Google News datasets, Filippova and Altun, 2013; MSR Abstractive Text Compression Dataset, Toutanova et al., 2016).

5.2 Approach 1: WordNet as Booster

Baseline methods are useful, but they are not developed for revision tasks after all. To replace a verb or noun phrase with a single word, we leverage word glosses in public dictionaries, *i.e.*, WordNet (Fellbaum, 2010). Word semantics are close to semantics in their glosses. This feature is usually used to improve word embedding (Bosc and Vincent, 2018) or evaluate analogy of word embedding (Mikolov et al., 2013). We use this feature to replace a verb or noun phrase with a single word.

We create data samples using WordNet and a language modelling corpus. For each sentence s in the corpus, we use WordNet vocabulary glosses to inflate it and obtain s'. Resulted parallel data approximate phrase replacement in sentence revision.

We first pick a unigram u, one of nouns, verbs, adjectives, or adverbs in s. At the same time, we avoid common words, *e.g.*, "*old*", or collocations and compounds, *e.g.*, "*united*" in "*United Kingdom*". Next, we apply Lesk's dictionary-based word sense disambiguation (WSD) algorithm (Lesk, 1986) on u and s to get gloss g. Then, we parse s and g to obtain respective dependency trees T_s and T_g; r_g denotes root node in T_g. Usually, if u is a noun, r_g is a noun, and if u is an adjective, r_g is a verb. Eight $u \rightarrow r_g$ patterns account for over 90% of the WordNet vocabulary (Table 5). In Algorithm 1, we modify dependency trees (T_s and T_g) according to the eight patterns. The remaining six patterns are NOUN→VERB, ADJ(-S) $\rightarrow$ ADJ, ADJ $\rightarrow$ ADP, ADJ $\rightarrow$ VERB, and ADV $\rightarrow$ ADP.

Finally, we filter and post-process synthesized sentences. We parse s' again and compare it with

Algorithm 1 Rule-based Gloss Substitution

Require: T_s, T_g **return** s'
  Copy-children(from u, to r_g)
  Locate h_u ▷ head node of u
  Delete (u with children, from T_s)
  if $u \in$ NOUN **then**
      Insert-child-node (r_g with children, to h_u)
      if $r_g \in$ VERB **then**
          $u \leftarrow$ Gerund(u)
      end if
      Correct inflections (singular and plural forms)
      Remove duplicate determiners
  else if $u \in$ VERB **then**
      Insert-child-node (r_g with children, to h_u)
      Correct inflections (person and tense)
      Add/Remove prepositions according to verb transitivity
  else
      Insert-right-child-node (r_g with children, to h_u) ▷ Post attributive
  end if
  $s' \leftarrow$ Linearize(T_s)

---

the dependency tree from which s' is linearized. We drop those with more than three mismatches, or with accuracy lower than 0.9. We "smooth" synthesized sentences with parroting[8], to mitigate overfitting. We also drop those sharing low semantic similarity (BERTScore ≤ 0.82) with original s.

We take the first 0.2 million sentences from WikiText-103 corpus (Merity et al., 2017) and around 71 thousand data points after filtration are available to train the BART base model.

5.3 Approach 2: Multi-Task Learning

Each dataset in baselines and Approach 1 handles part of task. However, sentence compression or simplification does not emphasize complete information retention; paraphrase generation hardly encourages deletion; synthetic data limit editing scope because word glosses are limited. We hypothesize that mixing the good samples among these datasets could more closely approximate the revision task. Therefore, we adjust datasets again. We keep every sample in MSR as it is small (21,145, see Appendix). Semantic similarity lower bound for sentence compression and simplification datasets is set at BERTScore = 0.9. For ParaNMT, we discard samples with less than 10 words. As a result, ablation of mixed and shuffled data samples shows that a mixture of MSR, filtered ParaNMT, and synthetic WordNet dataset leads to the strongest approach. This approach uses transfer learning from multiple datasets to learn revising

---

strategies such as deletion and phrase replacement.

5.4 Experiment and Result

Table 2 shows test results in each category. Our approach 2 has the highest overall score and is more robust than baseline models on category I, IV, and VII. The same architecture trained only on MSR outperforms any other baseline for deletion (category I) and ranks second for replacement (category II). The top-ranked baseline for replacement (category II) is trained on ParaNMT. In category V, the model trained on WordNet scores highest, slightly outperforming other baselines. Trends in category III, IV, VI, VII are less clear. Datasets Gigaword, Google News, and WikiSmall may be quite different from the benchmark corpus, and thus models trained on these datasets do not score well.

Our approach 2 suffers from two shortcomings common to all baselines. First, the model relies on transfer learning from MSR and ParaNMT and struggle to rewrite (category III) or to handle composite wordiness (category V, VI, VII). Second, the approach 2 outputs score worse than the input text on many metrics in many categories, especially on category III. These shortcomings suggest challenges in revision. We take the 5th and 95th percentile from all 536 samples to qualitatively illustrate the best proposed approach in Table 3. Apart from samples in Concise-536, Figure 1 shows an arbitrary sentence by non-English native speakers (Chen et al., 2020). The proposed revisioner removes repetition and unnecessary prepositional phrases, illustrating its potential in academic writing.

For human evaluation, we adopt an approach similar to Hsu et al. (2018); Zhang et al. (2020a); Ravaut et al. (2022). We (1) rank the samples by overall automatic evaluation on the model in descending order; (2) divide the examples in *each category* into two buckets; (3) randomly pick one example from each bucket. For each picked sample, we ask three graduate students (IELTS 7.0 or equivalent) to rank the predictions of seven systems, and the average ranking of each system is shown in **H** column in Table 2.

For top three systems, human evaluators then assess information retention (ρ) and wordiness (ω), since system outputs are in good syntax. Particularly, human assessment on wordiness engages the Paramedic Method (Lanham and Stodel, 1992) to highlight the wordy part and $\omega = (\text{\# wordy words})$

Methods	I	II	III	IV	V	VI	VII	All	H	ρ	ω
ParaNMT (Wieting and Gimpel, 2018)	0.46	**0.62**	0.46	*0.53*	0.44	0.45	*0.38*	0.55	3.40		
MSR (Toutanova et al., 2016)	*0.74*	0.58	0.44	0.51	0.41	0.44	0.37	0.57	*2.79*	0.78	**0.40**
G. News (Filippova and Altun, 2013)	*0.61*	0.46	0.39	0.40	0.35	0.39	0.33	0.48	5.74		
Gigaword (Rush et al., 2015)	0.30	0.29	0.28	0.31	0.25	0.29	0.23	0.29	6.74		
WikiSmall (Zhang and Lapata, 2017)	0.70	0.59	**0.48**	0.52	0.44	**0.46**	*0.38*	0.57	3.31		
Our Approach 1	0.70	0.60	*0.47*	0.52	**0.46**	*0.45*	0.37	*0.58*	2.79	**0.99**	0.47
Our Approach 2	**0.75**	0.60	*0.47*	**0.55**	0.40	*0.45*	**0.41**	**0.59**	**2.62**	*0.82*	*0.41*

Table 2: We average BLEU, METEOR, ROUGE-2-F1, SARI, Parsed relation F1, BERTScore-F1, *and* (negative) translation edit rate of (pre-)baseline methods. The most favorable score in each column is in bold, the second most favorable in italics. This table estimates the strengths and weaknesses of each variants. System ranking from human evaluation (**H**), information retention (ρ), and wordiness (ω) are presented in the right-most columns.

Category	Reference(s)	Prediction
5th percentile	Bob ~~provided an explanation of~~ explained the computer to his grandmother.	Bob ~~provided an explanation of~~ explained the computer to his grandmother.
95th percentile	~~Rather than taking the bull by the horns,~~ she ~~was quiet as a church mouse~~ avoided confrontation by remaining silent.	~~Rather than taking the bull by the horns,~~ she was quiet as a church mouse.

Table 3: Well/poorly revised samples in the corpus. Shorter sentences that require simpler actions are perfectly revised. Rewriting clichés is difficult, in which case the approach tends to use deletion.

/ (# all words). The model trained adapted Word-Net data preserves information better, which also accounts for its good human ranking.

We observe general correlation between automatic score ranking and human evaluation ranking. However, information retention is not sufficiently represented by semantic similarity scores like BERTScore. These findings suggest further investigation on the evaluation scheme of this task.

6 Discussion

Comparing the proposed revisioner's effectiveness for different categories, we understand deleting and replacing are much easier sub-tasks than rewriting is. The former two actions, especially deletion, are less ill-posed, while rewriting is open. Still, revision for concision requires an algorithm that is able to use all three actions in combination. Its goal is to resolve all seven categories of cases, marking distinction between revision and other tasks such as sentence compression.

We use seven metrics to estimate a revisioner's effectiveness, since each metric has its shortcomings. For example, METEOR does not adequately penalize nominalization, and thus wordy input texts typically score higher on METEOR than algorithm outputs. More targeted metrics for this task, including reference-free structural metrics (Sulem et al., 2018), might help. We do not include word counts. Although concision is marked by brevity and wordiness often correlates to high word count, concise writing does not always require the fewest words (PU). Optimizing a lower word count may be misleading even if it is constrained to zero information loss (Siddharthan, 2006). For example, abusing pronouns and ellipses can result in shorter sentences that are harder to read.

Transferring knowledge from other tasks to approximate revising is a stopgap measure. Specialized revising methods exist, *e.g.*, the Paramedic Method (Lanham and Stodel, 1992). Automated specialized methods may be more efficient.

7 Conclusion

We formulate sentence-level revision for concision as a constrained paraphrase generation task. The revision task not only requires semantics preservation as in usual paraphrasing tasks, but also specifies syntactic changes. A revised sentence is free of wordiness and as informative. Revising sentences is challenging and requires coordinated use of delete, replace, and rewrite. To benchmark revising algorithms, we collect 536 sentence pairs before and after revising from 72 college writing centres. We then propose a Seq2Seq revising model and evaluate it on this benchmark. Despite scarcity of training data, the proposed approaches offer promising results for revising academic texts. We believe this corpus will drive specialized revision algorithms that benefit both authors and readers.

Ethical considerations

The release of Concise-536 is intended only for "not-for-profit" educational purposes or private research and study in accordance with the Copyright Act 1994; all original text content is acknowledged as the property of each educational institution. All text content in Concise-536 (and the 120-point validation split) are public, and our release details their original links, thus making the release no different from a list of outbound links.

Limitations

The transfer of knowledge from other tasks to the rough revision is an emergency solution. There are specialised revision methods. For example, automating the Paramedic method (Lanham and Stodel, 1992) could possibly lead to a more efficient revisioner.

Acknowledgements

We would like to thank the reviewers for thoughtful comments and efforts to improve our manuscript.

References

Adam and Ben Long. 2013. Hemingway editor.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Tom Bosc and Pascal Vincent. 2018. Auto-encoding dictionary definitions into consistent word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1522–1532, Brussels, Belgium. Association for Computational Linguistics.

Tien-Cuong Bui, Van-Duc Le, Hai-Thien To, and Sang Kyun Cha. 2021. Generative pre-training for paraphrase generation by representing and predicting spans in exemplars. In *2021 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 83–90. IEEE.

Ruisheng Cao, Su Zhu, Chenyu Yang, Chen Liu, Rao Ma, Yanbin Zhao, Lu Chen, and Kai Yu. 2020. Unsupervised dual paraphrasing for two-stage semantic parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6806–6817, Online. Association for Computational Linguistics.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2003. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Current and new directions in discourse and dialogue*, pages 85–112. Springer.

John Carroll, Guido Minnen, Darren Pearce, Yvonne Canning, Siobhan Devlin, and John Tait. 1999. Simplifying text for language-impaired readers. In *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 269–270, Bergen, Norway. Association for Computational Linguistics.

R. Chandrasekar, Christine Doran, and B. Srinivas. 1996. Motivations and methods for text simplification. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Kai Chen, Haoqi Zhu, Leiming Yan, and Jinwei Wang. 2020. A survey on adversarial examples in deep learning. *Journal on Big Data*, 2(2):71.

Sumit Chopra, Michael Auli, and Alexander M. Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98, San Diego, California. Association for Computational Linguistics.

James Clarke and Mirella Lapata. 2006a. Constraint-based sentence compression: An integer programming approach. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 144–151, Sydney, Australia. Association for Computational Linguistics.

James Clarke and Mirella Lapata. 2006b. Models for sentence compression: A comparison across domains, training requirements and evaluation measures. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 377–384, Sydney, Australia. Association for Computational Linguistics.

Trevor Cohn and Mirella Lapata. 2008. Sentence compression beyond word deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 137–144, Manchester, UK. Coling 2008 Organizing Committee.

Commnet. The guide to grammar and writing.

William Coster and David Kauchak. 2011. Simple English Wikipedia: A new text simplification task. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 665–669, Portland, Oregon, USA. Association for Computational Linguistics.

Marie-Catherine De Marneffe and Christopher D Manning. 2008. Stanford typed dependencies manual. Technical report, Technical report, Stanford University.

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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2013. Paraphrase-driven learning for open question answering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1608–1618, Sofia, Bulgaria. Association for Computational Linguistics.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

Katja Filippova, Enrique Alfonseca, Carlos A. Colmenares, Lukasz Kaiser, and Oriol Vinyals. 2015. Sentence compression by deletion with LSTMs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 360–368, Lisbon, Portugal. Association for Computational Linguistics.

Katja Filippova and Yasemin Altun. 2013. Overcoming the lack of parallel data in sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1481–1491, Seattle, Washington, USA. Association for Computational Linguistics.

The Writing Center George Mason University. 2021. Writing concisely.

Chaitra V. Hegde and Shrikumar Patil. 2020. Unsupervised paraphrase generation using pre-trained language models. *CoRR*, abs/2006.05477.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701.

Wan-Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. 2018. A unified model for extractive and abstractive summarization using inconsistency loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 132–141, Melbourne, Australia. Association for Computational Linguistics.

Hongyan Jing. 2000. Sentence reduction for automatic text summarization. In *Sixth Applied Natural Language Processing Conference*, pages 310–315, Seattle, Washington, USA. Association for Computational Linguistics.

Hidetaka Kamigaito, Katsuhiko Hayashi, Tsutomu Hirao, and Masaaki Nagata. 2018. Higher-order syntactic attention network for longer sentence compression. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1716–1726, New Orleans, Louisiana. Association for Computational Linguistics.

Hidetaka Kamigaito and Manabu Okumura. 2020. Syntactically look-ahead attention network for sentence compression. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8050–8057. AAAI Press.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1328–1338, Austin, Texas. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. 2016. Improving sentence compression by learning to predict gaze. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1528–1533, San Diego, California. Association for Computational Linguistics.

Kevin Knight and Daniel Marcu. 2000. Statistics-based summarization - step one: Sentence compression. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on on Innovative Applications of Artificial Intelligence, July 30 - August 3, 2000, Austin, Texas, USA*, pages 703–710. AAAI Press / The MIT Press.

Mahnaz Koupaee and William Yang Wang. 2018. Wikihow: A large scale text summarization dataset. *CoRR*, abs/1810.09305.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*,

pages 1224–1234, Copenhagen, Denmark. Association for Computational Linguistics.

Richard A Lanham and James Stodel. 1992. *Revising prose*. Macmillan Publishing Company.

Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation, SIGDOC 1986, Toronto, Ontario, Canada, 1986*, pages 24–26. ACM.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Zhouhan Lin, Minwei Feng, Cícero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Ryan McDonald. 2006. Discriminative sentence compression with soft syntactic evidence. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 297–304, Trento, Italy. Association for Computational Linguistics.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Merriam-Webster. Concise definition.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Monash University (MON) Writing Department MON. 2020. Writing clearly, concisely and precisely.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Kostiantyn Omelianchuk, Vipul Raheja, and Oleksandr Skurzhanskyi. 2021. Text Simplification by Tagging. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 11–25, Online. Association for Computational Linguistics.

Paul Over, Hoa Dang, and Donna Harman. 2007. Duc in context. *Information Processing & Management*, 43(6):1506–1520.

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*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Purdue University (PU) Writing Lab PU. Concision.

Avinesh P.V.S and Christian M. Meyer. 2019. Data-efficient neural text compression with interactive learning. 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)*, pages 2543–2554, Minneapolis, Minnesota. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Randy Rambo. 2019. English composition 1.

Mathieu Ravaut, Shafiq Joty, and Nancy F. Chen. 2022. Summareranker: A multi-task mixture-of-experts re-ranking framework for abstractive summarization. *CoRR*, abs/2203.06569.

Stefan Riezler, Tracy H. King, Richard Crouch, and Annie Zaenen. 2003. Statistical sentence condensation using ambiguity packing and stochastic disambiguation methods for Lexical-Functional Grammar. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 197–204.

Vasile Rus, Rajendra Banjade, and Mihai Lintean. 2014. On paraphrase identification corpora. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2422–2429, Reykjavik, Iceland. European Language Resources Association (ELRA).

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Evan Sandhaus. 2008. The new york times annotated corpus.

Advaith Siddharthan. 2006. Syntactic simplification and text cohesion. *Research on Language and Computation*, 4(1):77–109.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Engineering Stanford University. Writing style guide.

Keh-Yih Su, Ming-Wen Wu, and Jing-Shin Chang. 1992. A new quantitative quality measure for machine translation systems. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018. Semantic structural evaluation for text simplification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 685–696, New Orleans, Louisiana. Association for Computational Linguistics.

Kristina Toutanova, Chris Brockett, Ke M. Tran, and Saleema Amershi. 2016. A dataset and evaluation metrics for abstractive compression of sentences and short paragraphs. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 340–350, Austin, Texas. Association for Computational Linguistics.

University of Arkansas at Little Rock (UALR) Writing Center UALR. Somes steps toward concise writing.

University of North Carolina (UNC) Chapel Hill Writing Center UNC. 2021. Writing concisely.

Massey University (UNZ) The Online Writing & Learning Link (OWLL) UNZ. Writing concisely.

Learning Hub UOA, The University of Auckland (UOA). Concise writing.

The Graduate Writing Center URI, The University of Rhode Island (URI). 2019. Concise writing: Tips and tricks.

John Wieting and Kevin Gimpel. 2018. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Melbourne, Australia. Association for Computational Linguistics.

Walden University (WU) Writing Center WU. Scholarly voice: Writing concisely.

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*, 4:401–415.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Mark Yatskar, Bo Pang, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2010. For the sake of simplicity: Unsupervised extraction of lexical simplifications from Wikipedia. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 365–368, Los Angeles, California. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

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*, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.

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 2010)*, pages 1353–1361, Beijing, China. Coling 2010 Organizing Committee.

A Linguistic Rules in Revising for Concision

We collate and present a set of practical linguistic rules for concise sentence revision, which we synthesize based on guidelines from writing centres at numerous major universities and educational institutes. Table 4 illustrates how wordiness can be fine-grained, and what action is required once a wordiness is identified (UNC, 2021; PU; MON, 2020).

B Technical Difficulties in Reference-free Revision Evaluation

Had we chosen not to follow Papineni's viewpoint, reference-free evaluation is the way to go. However, it is technically not trivial to use programs to detect wordiness or syntax errors these days (See Section 2.3), let alone detect semantic similarity. Progress in sentence embedding (Lin et al., 2017) and semantic textual similarity (Yang et al., 2019) enables meaning comparison between sentences, but relying on one developing system to evaluate another is risky. Moreover, information delivered by a sentence is sometimes beyond its textual meaning. Concise writing can suggest eliminating first-person narratives; *e.g.*, "*I feel that the study is significant*" is revised to "*The study is significant*" (WU). Here, the first-person statement

Wordiness identified	Action
Weak modifiers (qualifiers / intensifiers)	Delete
Redundant pairs	
Grouped synonyms	
Stock phrases	
Unnecessary hedging	
Implied information	
Yourself	
Informal language	Replace
Vague pronoun references	
Possessive constructions using "of"	
Prepositional phrases	
All-purpose nouns	
Vague Swamp	
Fancy words	
Helping verbs ("to be" verbs, "be" + adjective)	
Adjective-noun pairs	
Phrasal verbs	
Verb-adverb pairs	
Nominalisation / noun strings	
Cliches and Euphemisms	
Empty phrases	
Expletive constructions	
long sentences (>25 words)	Rewrite
Running starts (with "there / it" + "be")	
Long opening phrases / clauses	
Needless transitions	
Interrupted subjects and verbs	
Interrupted verbs and objects	
Negatives (opposite to affirmatives)	
and anything violating:	
A blend of active and passive verbs	
Elliptical constructions / parallelism	
Only one main idea per sentence	

Table 4: Revising rules collated from college writing centers. Three actions are available. Redundancy can be deleted; short, specific, concrete and stronger expressions shall replace vague ones; sentences should be rewritten if neither deleting nor replacing helps.

used to be the main clause, and removing it will shift sentence embedding. Nevertheless, in academic writing, these two sentences deliver identical information.

C Balance between Syntax, Information, and Wordiness

The coefficient α tells how much syntax overweighs information, or information overweighs reduced wordiness. Empirically, minimum of α can be around the word count in a standard sentence. In other words, even if a single key word is missing, the decrease in ρ is bigger than the increase in $1 - \omega$.
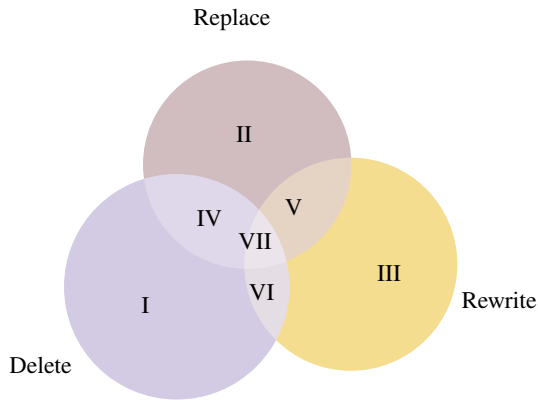
Figure 2: Revising a sentence can involve either one of the three strategies (category I, II, III), or a combination of them (category IV, V, VI, VII).

D   Explaining Categories in the Corpus

There are seven categories, as seen in Figure 2. Note that although three actions (delete, replace, rewrite) are put side-by-side, they are with different levels of flexibility. In fact, every revision made with deleting can be done through replace, *e.g.*, the fourth example in Table 9, "*fell down*" could be replaced to "*fell*", but we would simply consider the cheapest revision, which is to delete "*down*"[9]. Similarly, rewriting is even more expensive and ambiguous. Therefore, our rule of Occam's razor is that only when a cheaper revision fails, will we use a more expensive one.

Here, we give examples of which category a revision corresponds to. Indeed, many sentence revisions are categorized in original websites. See the two examples from Purdue Writing Lab (PU) below. The strategies applied are to "eliminate words that explain the obvious or provide excessive detail" (category I) and to "replace several vague words with more powerful and specific words" (category II), respectively.

Corpus Example D.1 (category I). Imagine ~~a mental picture of~~ someone ~~engaged in the intellectual activity of~~ trying to learn ~~what~~ the rules ~~are for how to play the game~~ of chess.

Corpus Example D.2 (category II). The politician ~~talked about several of the merits of~~ touted after-school programs in his speech

For revisions not categorized in sources, we first align the segments of a pair of sentences by their meaning, as seen in Figure 1. This is intuitively straightforward when the revised sentence

---

[9]"*fell*" means "*fell down*", as one never "*fell up*".

is given[10].

Then, we determine the actions to revise. For example, in the fourth example (category IV) in Table 9, we find that we cannot delete any words in "*due to the fact that*" without violating the second and third components in Definition 2.4. Thus, we have to put some more concise conjunction to take its place, *i.e.*, "*because*".

Another example is the sixth one (category IV) in Table 10. Though it looks that the entire wordy sentence can only be written to reach the concise form, a cheaper revision is actually to first delete some redundancy, *e.g.*, "*sent ~~to you~~ by us*", and then rewrite the necessary part.

Whether the subject and predicate of a sentence (clause) is changed together determines the border between replacing and rewriting. In the fifth example (category IV) in Table 9, "*it was necessary*" is aligned to "*had to*", and "*us*" to "*we*". However, we cannot change either of them individually without violating the third component in Definition 2.4. Therefore, when two or more replacements intertwine, we rewrite.

E   Explaining Rule-based Gloss Substitution

A demonstration of Algorithm 1 is shown in Fig.3, where a verb that appears in the past participle is replaced. By running this rule-based gloss replacement multiple times, we can recursively expand a sentence because the words used in a gloss have their associated glosses (Bosc and Vincent, 2018). Table 5 describes $u \to r_g$ in the WordNet vocabulary.

F   Details in Datasets Used to Train baselines

We prepare training samples by adjusting data from paraphrase generation, sentence simplification, or sentence compression. ParaNMT (Wieting and Gimpel, 2018) contains over five million paraphrase pairs annotated from machine translation tasks; we sort each pair by sentence length. This is a rough approximation, since shorter sentences are not necessarily more concise. Google News datasets (News, Filippova and Altun, 2013) contains 0.2 million pairs of sentences, where the longer one is the leading sentence of each article, and the shorter one is a subsequence of the

---

[10]If the revised sentences were not from a trustworthy site, this process could have been less intuitive.
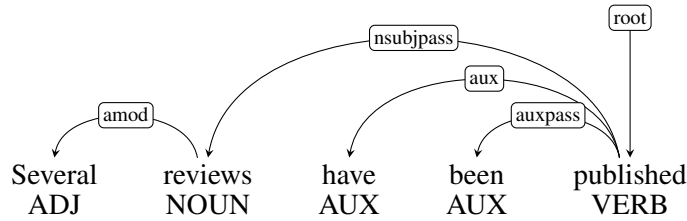
(a) Sentence "*Several reviews have been published*" and its dependency tree. We expand the word "*publish*" below.
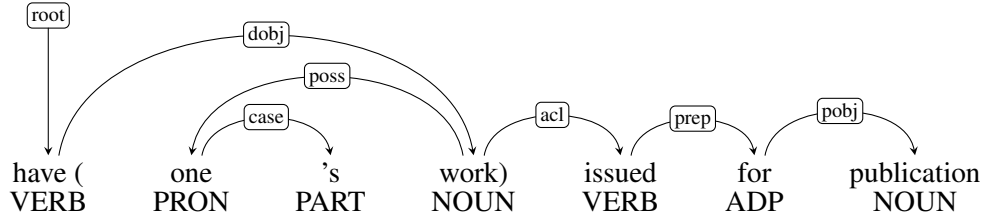
(b) Gloss of "*publish*" from WordNet(Fellbaum, 2010); the root node is the verb "*have*".
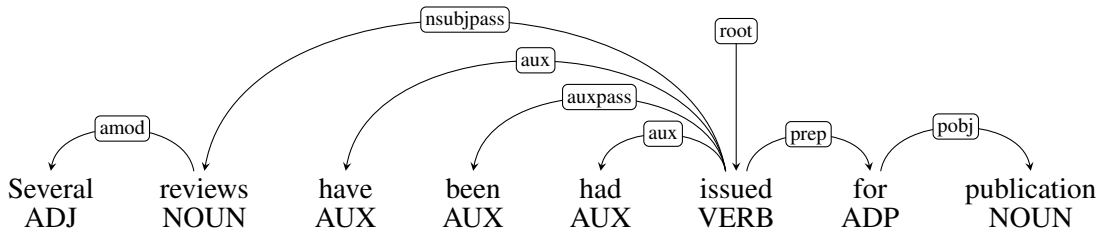
(c) Synthesized sentence with the first stage of our approach; r_g node "*have/had*" is grafted onto the original sentence in (a).

Figure 3: Demonstration of dependency tree grafting in sentence synthesis. The dependency in (c) is obtained by re-parsing the synthesized sentence. As we can see, the POS tag of "*have/had*" has changed from a verb to an auxiliary word, and the synthesized sentence is still syntactically and semantically correct, which shows that dependency changes may be unavoidable in the process of sentence synthesis. We also dealt with inflections to reduce grammatical errors.

longer one. Gigaword (Rush et al., 2015) contains four million pairs of article headline and the first sentence. Although these datasets are mainly for generating news headlines (P.V.S and Meyer, 2019), they approximate the deletion aspect of sentence revision. MSR Abstractive Text Compression Dataset (Toutanova et al., 2016) contains six thousand sentence pairs from business letters, newswire, journals, and technical documents sampled from the Open American National Corpus[11]; humans rewrite sentences at a fixed compression ratio. WikiSmall (Zhang and Lapata, 2017) contains sentence pairs from Wikipedia articles and corresponding Simple English Wikipedia. We adopt training splits of these datasets, and Table 6 lists their sizes.

G   Random Sample Selection in Human Evaluation

```
import random
```

```
random.seed(0)
for k in [169, 116, 153, 42, 33, 14, 9]:
    print(random.randint(0, k//2))
    print(random.randint(k//2, k))
```

H   Evaluation on Individual Metrics

For each sample in the benchmark corpus, we compute individual metric score for its best-revised sentence and average the corpus ranking of its individual metric scores to obtain the final ranking for that sample. Table 9 lists the well-treated samples (at the third percentile) in each category. Table 10 lists the cases that were not well resolved (at the 97th percentile).

Figure 4 shows the difficulty of revising sentences for each category. The data in Figure 4, while demonstrating strengths and weaknesses of the proposed approach, can also serve as an approximation of the difficulty of the corpus itself. The proposed approach is better at deleting and replacing than rewriting due to heavy reliance on transfer learning.

---

[11]https://www.anc.org/data/oanc

POS	ADJ	ADJS	ADV	NOUN	VERB
VERB	**3627**	**6354**	221	**3349**	**11586**
DET	1	6	4	1594	0
ADJ	1053	**2825**	50	544	316
NOUN	155	405	57	**73527**	1739
CCONJ	0	0	0	24	0
PUNCT	0	1	0	6	0
PART	0	5	4	4	6
ADV	10	84	235	37	52
ADP	**2615**	972	**3019**	222	29
AUX	5	5	4	108	1
PRON	0	3	2	1516	1
SCONJ	4	10	14	3	10
PROPN	0	6	0	534	15
X	2	2	2	16	19
NUM	1	16	8	658	0
INTJ	1	1	3	13	12
SYM	0	0	0	0	0

Table 5: Part-of-speech (POS) tags for a word w and its corresponding r_g. Representation of POS tags follows the Stanford typed dependencies manual (De Marneffe and Manning, 2008) (except for ADJ-S, which stands for 'adjective satellite' in WordNet (Fellbaum, 2010)). POS tags of r_g are closely related to the POS tags of w, and we bold the pairs that appear frequently. In particular, among nearly 117,000 word-gloss ($w \rightarrow r_g$) pairs, NOUN $\rightarrow$ NOUN is most frequent, accounting for more than three fifths. We have now studied the eight most frequently occurring pairs.
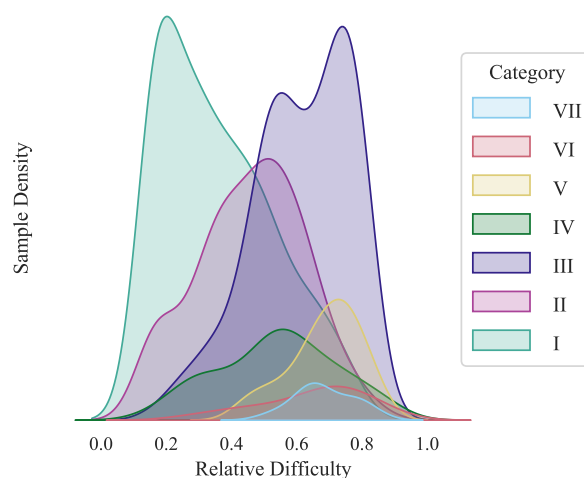
Figure 4: Difficulty faced by the proposed approachs when dealing with sentences from different categories. This difficulty is relative to other samples in the corpus of 536 sentences. Deletion (category I) is the least challenging. The most challenging samples are most likely from category III. Handling sentences requiring more than one revising strategies (category IV-VII) is usually more challenging.

Dataset	Size
MSR (2016)	21,145
ParaNMT (2018)	5,306,522
Google News (G. News 2013)	200,000
Gigaword (2015)	3,803,957
WikiSmall (2017)	89,042
Approach 2 fine-tuning set (Section 5.3)	182,330

Table 6: Sample numbers of training sets. MSR dataset has multiple references, we take each reference as a sample point. The mixed fine-tuning set in Section 5.3 is composed of 89,712 samples from ParaNMT, 21,145 from MSR, and 71,473 from our synthesized dataset from WordNet.

	BL	M	R	S	P	BS	T
ParaNMT	.40	.78	.56	.49	.55	.96	.53
MSR	.55	.86	.69	.62	.73	**.97**	.39
News	.36	.67	.56	.52	.62	.95	.51
Gigaword	.04	.27	.19	.25	.16	.89	.92
WikiSmall	.47	**.91**	.65	.59	.63	.95	.82
Approach 1	.48	.90	.65	.59	.66	.94	.82
Approach 2	**.57**	.87	**.71**	**.66**	**.74**	**.97**	**.37**

(a) Category I, 169 / 536, Delete

	BL	M	R	S	P	BS	T
ParaNMT	**.38**	.75	**.54**	**.54**	**.51**	**.97**	**.50**
MSR	.33	.73	.51	.45	.47	.96	.56
News	.17	.56	.39	.35	.37	.94	.64
Gigaword	.01	.26	.17	.28	.15	.89	.92
WikiSmall	.33	.80	.51	.48	.44	.95	.92
Approach 1	.36	**.80**	**.54**	.51	.46	.95	.91
Approach 2	.36	.76	.53	.50	.50	.96	.53

(b) Category II, 116 / 536, Replace

	BL	M	R	S	P	BS	T
ParaNMT	.16	.60	.33	.44	.28	**.94**	.94
MSR	.15	.56	.31	.38	.28	.93	.92
News	.10	.44	.26	.37	.25	.92	.85
Gigaword	.03	.20	.12	.38	.09	.88	1.00
WikiSmall	**.19**	.66	.35	**.45**	.29	.93	1.26
Approach 1	.17	.65	**.36**	.43	.29	.93	1.27
Approach 2	.18	.60	.35	.42	**.30**	**.94**	**.91**

(c) Category III, 153 / 536, Rewrite

	BL	M	R	S	P	BS	T
ParaNMT	.27	.68	.40	**.48**	.41	.94	1.22
MSR	.26	.63	.39	.44	.39	.94	1.09
News	.12	.41	.29	.37	.31	.92	**.77**
Gigaword	.03	.24	.16	.35	.16	.89	.86
WikiSmall	.24	.71	.39	.46	.37	.93	1.69
Approach 1	.23	**.72**	.39	.44	.39	.92	1.70
Approach 2	**.32**	.67	**.44**	.47	**.43**	**.95**	.94

(d) Category IV, 42 / 536, Delete + Replace

	BL	M	R	S	P	BS	T
ParaNMT	.16	.55	.27	**.44**	.26	**.94**	1.18
MSR	.14	.49	.28	.37	.24	.93	1.05
News	.08	.34	.21	.35	.22	.92	**.85**
Gigaword	.01	.15	.09	.34	.06	.87	.90
WikiSmall	.17	.58	.30	.40	.27	.93	1.40
Approach 1	**.20**	**.59**	**.31**	.43	**.29**	.93	1.37
Approach 2	.14	.49	.26	.35	.25	.93	1.04

(e) Category V, 33 / 536, Replace + Rewrite

	BL	M	R	S	P	BS	T
ParaNMT	.21	.56	.30	**.43**	.27	**.93**	1.35
MSR	.19	.53	.30	.40	.30	**.93**	1.23
News	.12	.39	.26	.37	.25	.92	**.81**
Gigaword	.04	.20	.12	.38	.10	.87	.87
WikiSmall	.17	.62	**.31**	.43	**.30**	.92	1.92
Approach 1	.16	**.62**	**.31**	.41	.29	.92	1.96
Approach 2	**.22**	.55	**.31**	.41	**.30**	**.93**	1.23

(f) Category VI, 14 / 536, Delete + Rewrite

	BL	M	R	S	P	BS	T
ParaNMT	.06	.50	.17	.40	.20	.92	1.61
MSR	.06	.49	.17	.39	.17	.92	1.34
News	.03	.28	.15	.38	.21	.91	**.79**
Gigaword	.00	.11	.04	.36	.02	.86	.95
WikiSmall	**.08**	.54	.20	.38	.18	.91	2.02
Approach 1	.04	.53	.18	.39	.17	.91	1.96
Approach 2	**.08**	**.55**	**.23**	**.43**	**.24**	**.93**	1.18

(g) Category VII, 9 / 536, Delete + Replace + Rewrite

	BL	M	R	S	P	BS	T
ParaNMT	.29	.69	.44	.48	.42	**.95**	.77
MSR	.32	.69	.48	.48	.47	**.95**	.71
News	.20	.53	.38	.41	.40	.93	.69
Gigaword	.03	.23	.15	.31	.13	.88	.94
WikiSmall	.31	.76	.48	.49	.43	.94	1.12
Approach 1	.31	**.76**	.49	.50	.45	.94	1.12
Approach 2	**.35**	.72	**.50**	**.51**	**.49**	**.95**	**.68**

(h) Overall

Table 7: BLEU (**BL**), METEOR (**M**) , ROUGE-2-F1 (**R**), SARI (**S**), Parsed relation F1 (**P**), BERTScore-F1 (**BS**), *and* translation edit rate (**T**) of pre-Approach 2s and Approach 2 method. Numbers are shown in categories. Smaller edit distance is more favorable. The most favorable score(s) in each column is bold. In category V, the model trained on Approach 1 has the highest scores on three metrics, slightly outperforming other pre-Approach 2s. In category III,IV, VI, VII, no particular pre-Approach 2 scores well on all metrics.

	W	R1	RL
ParaNMT	.65	.75	.71
MSR	.43	.85	.8
News	.54	.73	.7
Gigaword	.97	.39	.37
WikiSmall	.89	.82	.77
Approach 1	.96	.82	.77
Approach 2	**.42**	**.86**	**.81**

(a) Category I, 169 / 536, Delete

	W	R1	RL
ParaNMT	.61	**.73**	**.72**
MSR	.58	.71	.7
News	.66	.6	.59
Gigaword	.98	.37	.35
WikiSmall	.93	.71	.7
Approach 1	1.03	**.73**	**.72**
Approach 2	**.54**	**.73**	**.72**

(b) Category II, 116 / 536, Replace

	W	R1	RL
ParaNMT	1.04	.61	.50
MSR	.99	.60	.48
News	.88	.52	.43
Gigaword	1.03	.32	.29
WikiSmall	1.35	.63	.50
Approach 1	1.40	.63	.50
Approach 2	**.97**	.63	**.51**

(c) Category III, 153 / 536, Rewrite

	W	R1	RL
ParaNMT	1.28	.59	.58
MSR	1.14	.59	.57
News	**.81**	.48	.46
Gigaword	.90	.37	.32
WikiSmall	1.74	.58	.56
Approach 1	1.79	.59	.56
Approach 2	.97	**.63**	**.62**

(d) Category IV, 42 / 536, Delete + Replace

	W	R1	RL
ParaNMT	1.36	.52	.45
MSR	1.17	.51	.43
News	**.91**	.42	.36
Gigaword	.96	.28	.24
WikiSmall	1.58	.53	.44
Approach 1	1.59	**.55**	**.46**
Approach 2	1.15	.52	.43

(e) Category V, 33 / 536, Replace + Rewrite

	W	R1	RL
ParaNMT	1.54	.49	.41
MSR	1.37	.51	.41
News	**.87**	.45	.38
Gigaword	.89	.30	.30
WikiSmall	2.04	.51	.43
Approach 1	2.10	.51	.43
Approach 2	1.39	**.54**	**.45**

(f) Category VI, 14 / 536, Delete + Rewrite

	W	R1	RL
ParaNMT	1.64	.44	.35
MSR	1.38	.47	.35
News	**.81**	.38	.34
Gigaword	.97	.25	.20
WikiSmall	2.04	.44	.33
Approach 1	1.98	.45	.34
Approach 2	1.24	**.53**	**.43**

(g) Category VII, 9 / 536, Delete + Replace + Rewrite

	W	R1	RL
ParaNMT	.88	.66	.61
MSR	.76	.69	.63
News	**.72**	.59	.55
Gigaword	.98	.35	.33
WikiSmall	1.19	.69	.63
Approach 1	1.25	.70	.63
Approach 2	.73	**.71**	**.65**

(h) Overall

Table 8: Other metrics include word error rate (**W**), ROUGE-1-F1 (**R1**), *and* ROUGE-L-F1 (**RL**) . Numbers are shown in categories. Smaller edit distance is more favorable. The most favorable score(s) in each column is bold.

Category	Reference(s)	Prediction
I	~~The reason~~ she moved ~~is~~ because she was offered a better position.	~~The reason~~ she moved ~~is~~ because she was offered a better position.
II	We outlined the letters for the banner ~~in a careful way~~ carefully.	We outlined the letters for the banner ~~in a careful way~~ carefully.
III	~~The current focus of the medical profession is~~ The medical profession currently focuses on disease prevention.	~~The current focus of the medical profession is~~ The medical profession focuses on disease prevention.
IV	She fell ~~down due to the fact that~~ because she hurried.	She fell ~~down due to the fact that~~ because she hurried.
V	We didn't renew the contract with our supplier ~~on account of the fact that~~ because ~~it was necessary for us to~~ we had to find a cheaper ~~supplier~~ one.	We didn't renew the contract ~~with our supplier on account of the fact that~~ because it was necessary for us to find a cheaper supplier.
VI	Research is increasing in ~~the field of~~ nutrition and food science. *OR* Research within nutrition and food science is increasing ~~the field of nutrition and food science~~.	Research is increasing in ~~the field of~~ nutrition and food science.
VII	Joe found ~~that the fictional novel by~~ Alcott's book, Little Women, ~~conveyed a sense of emotion and feeling~~ both delightful and tragic.	Joe found that ~~the fictional novel by~~ Alcott's novel, Little Women, conveyed ~~a sense of~~ emotion and feeling.

Table 9: Well-revised samples in each category (about the third percentile). Shorter sentences that require simpler actions are perfectly revised. When sentence length increases, not all wordiness can be resolved. The baseline method tends to delete and replace, even if sentences need to be rewritten.

Category	Reference(s)	Prediction
I	~~I would call your attention to the fact that~~ our President, ~~who was~~ formerly the Governor of Arkansas, is ~~basically~~ a Southerner.	I would call your attention to the fact that our President ~~, who was formerly the Governor of Arkansas,~~ is a ~~basically~~ Southerner.
II	We have ~~conducted an investigation~~ investigated and ~~arrived at the conclusion~~ concluded.	We ~~have~~ conducted an investigation and arrived at the conclusion.
III	~~As far as my professor is concerned, the problem of wordiness is the thing she'd really like to see us involved with actually eliminating.~~ My professor wants us to focus on eliminating wordiness.	~~As far as my professor is concerned,~~ the problem of wordiness is the thing she'd ~~really~~ like to see us involved with ~~actually~~ eliminating.
IV	~~The existence of~~ computers ~~and computer technology has~~ have improved ~~commercial enterprise~~ business ~~and information exchange~~ communication.	The existence of computers and ~~computer~~ technology has ~~greatly~~ influenced commercial enterprise and information exchange.
V	~~It has come to my attention that~~ I have noticed ~~there is a vast proliferation of undesirable vegetation~~ many weeds growing ~~surrounding the periphery of~~ around ~~this facility~~ the building	It has come to my attention that there is a vast proliferation of undesirable vegetation ~~surrounding the periphery of this facility~~.
VI	~~Your statement in your letter that has the contention that the information submitted and sent to you by us contained certain inaccuracies and errors has prompted us to embark on a careful and thorough reevaluation of the information submitted, with the result that the original informative data has been determined to be accurate and correct in all instances and aspects of the information.~~ As you suggested, we have checked our information and confirmed its accuracy.	Your statement in your letter that ~~has the contention that~~ the information submitted and sent to you by us contained ~~certain~~ some inaccuracies and errors has prompted us to embark on a ~~careful and~~ thorough reevaluation of the information ~~submitted~~, with the result that the original informative data has been determined to be accurate and correct ~~in all instances and aspects of the information~~.
VII	~~In the event that~~ If you get ~~some~~ any information ~~concerning~~ about Mr. Smith ~~should be brought to your attention~~, ~~it should be forwarded via mail or courier or telephone to us~~ please contact us ~~in view of the possibility that~~ in case ~~the information may reveal any attempt on the part of Mr. Smith to depart from the United States~~ he tries to leave the country.	~~In the event that~~ If ~~some~~ any information concerning Mr. Smith should be brought to your attention, it should be forwarded via mail or courier or telephone to us in view of the possibility that the information may reveal any attempt ~~on the part of Mr. Smith~~ to depart from the United States.

Table 10: Badly-revised samples in each category (about the 97th percentile). These sentences are longer than sentences in Table 9. Informative part may be trimmed. Replacing nominalizations with verbs is hard. For severely wordy sentences (category VI, VII), the model fails to rewrite, and resorts to deletion. A lot of improvement is needed.