The Fewer Splits are Better: Deconstructing Readability in Sentence Splitting

Tadashi Nomoto
National Institute of Japanese Literature
Tachikawa, Tokyo 190-0014, Japan
nomoto@acm.org

Abstract

In this work, we focus on sentence splitting, a subfield of text simplification, motivated largely by an unproven idea that if you divide a sentence in pieces, it should become easier to understand. Our primary goal in this paper is to find out whether this is true. In particular, we ask, does it matter whether we break a sentence into two or three? We report on our findings based on Amazon Mechanical Turk.

More specifically, we introduce a Bayesian modeling framework to further investigate to what degree a particular way of splitting the complex sentence affects readability, along with a number of other parameters adopted from diverse perspectives, including clinical linguistics, and cognitive linguistics. The Bayesian modeling experiment provides clear evidence that bisecting the sentence leads to enhanced readability to a degree greater than when we create simplification by trisection.

1 Introduction

In text simplification, one question people often fail to ask is, whether the technology they are driving truly helps people better understand texts. This curious indifference may reflect the tacit recognition of the partiality of datasets covered by the studies (Xu et al., 2015) or some murkiness that surrounds the goal of text simplification.

As a way to address the situation, we examine a role of simplification in text readability, with a particular focus on sentence splitting. The goal of sentence splitting is to break a sentence into small pieces in a way that they collectively preserve the original meaning. A primary question we ask in this paper is, does a splitting of text affect readability? In the face of a large effort spent in the past on sentence splitting, it comes as a surprise that none of the studies put this question directly to people; in most cases, they ended up asking whether generated texts 'looked simpler' than the original unmodified versions (Zhang and Lapata, 2017), which of course does not say much about their readability. We are not even sure whether there was any agreement among people on what constituted simplification.

Another related question is, how many pieces should we break a sentence into? Two, three, or more? In the paper, we focus on a particular setting where we ask whether there is any difference in readability between two- and three-sentence splits. We also report on how good or bad sentence splits are that are generated by a fine-tuned language model, compared to humans'.

A general strategy we follow in the paper is to elicit judgments from people on whether simplification made a text anyway readable for them (Section 4), and do a Bayesian analysis of their responses to identify factors that may have influenced their decisions (Section 5).[1]

2 Related Work

Historically, there have been extensive efforts in ESL (English as a Second Language) to explore the use of simplification as a way to improve reading performance of L2 (second language) students. Crossley et al. (2014) presented an array of evidence showing that simplifying text did lead to an improved text comprehension by L2 learners as measured by reading time and and accuracy of their responses to associated questions. They also noticed that simple texts had less lexical diversity, greater word overlap, greater semantic similarity among sentences than more complicated texts. Crossley et al. (2011) argued for the importance of cohesiveness as a factor to influence the readability. Meanwhile, an elaborative modification of text was found to play a role in enhancing readability, which involves adding information

[1] We will make available on GitHub the data we created for the study soon after the paper's publication (they should be found under https://github.com/tnomoto).

Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022), pages 1 - 11
December 8, 2022 ©2022 Association for Computational Linguistics

to make the language less ambiguous and rhetorically more explicit. Ross et al. (1991) reported that despite the fact that it made a text longer, the elaborative manipulation of a text produced positive results, with L2 students scoring higher in comprehension questions on modified texts than on the original unmodified versions.

While there have been concerted efforts in the past in the NLP community to develop metrics and corpora purported to serve studies in simplification (Zhang and Lapata, 2017; Sulem et al., 2018a; Narayan et al., 2017; Botha et al., 2018; Niklaus et al., 2019; Kim et al., 2021; Xu et al., 2015), they fell far short of addressing how their work contributes to improving the text comprehensibility by readers. Part of our goal is to break away from a prevailing view that relegates the readability to a sideline.

3 Method

The data come from two sources, the Split and Rephrase Benchmark (v1.0) (SRB, henceforth) (Narayan et al., 2017) and WikiSplit (Botha et al., 2018). SRB consists of complex sentences aligned with a set of multi-sentence simplifications varying in size from two to four. WikiSplit follows a similar format except that each complex sentence is accompanied only by a two-sentence simiplification.[2] We asked Amazon Mechanical Turk workers (Turkers, henceforth) to score simplifications on linguistic qualities as well as to indicate whether they have any preference between two-sentence and three-sentence versions in terms of readability.

We randomly sampled a portion of SRB, creating test data (call it $\mathcal{H}$), which consisted of triplets of the form: $\langle S_0, A_0, B_0\rangle$, . . . , $\langle S_i, A_i, B_i\rangle$, . . . , $\langle S_m, A_m, B_m\rangle$, where S_i is a complex sentence, A_i a corresponding two-sentence simplification, and B_i its three-sentence version. While A alternates between versions created by BART and by human, B deals only with manual simplifications.[3] See Table 1 for a further explanation.

	BART	HUM
A (TWO-SENTENCE SPLIT)	113	108
B (THREE-SENTENCE SPLIT)	–	221

Table 1: A break down of $\mathcal{H}$. 113 of them are of type A (bipartite split) generated by BART-large; 108 are of type A created by humans. There were 221 of type B (tripartite split), all of which were produced by humans.

TRAIN	DEV
1,135,009 (989,944)	13,797(5,000)

Table 2: A training setup for BART. The data comes from SRB (Narayan et al., 2017) and WikiSplit (Botha et al., 2018). The parenthetical numbers indicate amounts of data that originate in WikiSplit (Botha et al., 2018).

Separately, we extracted from WikiSplit and SRB, another dataset $\mathcal{B}$ consisting of complex sentences as a source and two-sentence simplifications as a target (Table 2) i.e. $\mathcal{B} = \{\langle S'_0, A'_0\rangle, \ldots, \langle S'_n, A'_n\rangle\}$, to use it to fine-tune a language model (BART-large).[4] The fine-tuning was done using a code available at GitHub.[5]

A task (or a HIT in Amazon's parlance) we asked Turkers to do was to work on a three-part language quiz. The initial problem section introduced a worker to three short texts, corresponding to a triplet $\langle S_i, A_i, B_i\rangle$; the second section asked about linguistic qualities of A_i and B_i along three dimensions, *meaning*, *grammar*, and *fluency*; and in the third, we asked two comparison questions: (1) whether A_i and B_i are more readable than S_i, and (2) which of A_i and B_i is easier to understand.

Figure 1 gives a screen capture of an initial section of the task. Shown Under **Source** is a complex sentence or S_i for some i . **Text A** and **Text B** correspond to A_i and B_i, which were displayed in a random order.

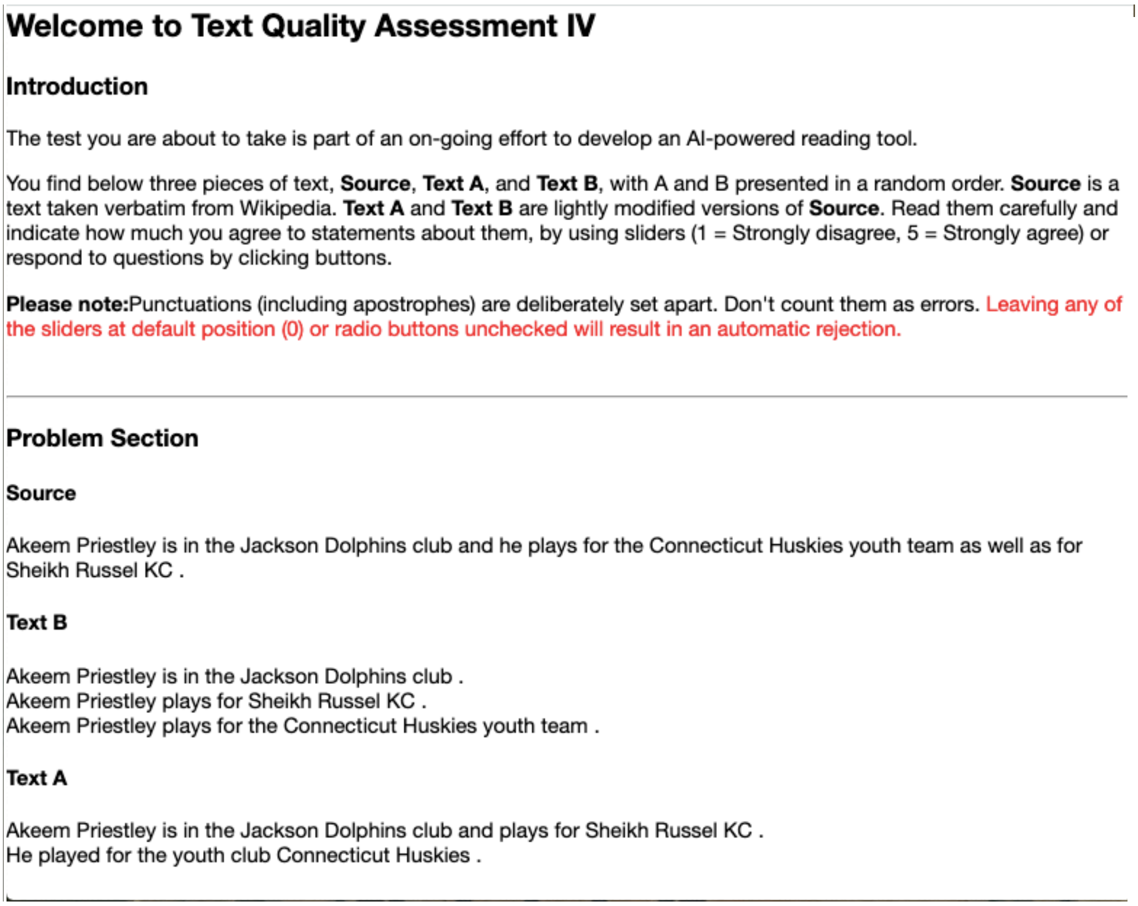
In total, there were 221 HITs (Table 1), each administered to seven people. All of the participants were self-reported native speakers of English with a degree from college or above. The participation was limited to residents in US, Canda, UK, Australia, and New Zealand.

[2] We used WikiSplit, together with part of SRB, exclusively to fine tune BART to give a single split (bipartite) simplification model, and SRB to develop test data to be administered to humans for linguistic assessments. SRB was derived from WebNLG (Gardent et al., 2017) by making use of RDFs associated with textual snippets to assemble simplifications.

[3] HSplit (Sulem et al., 2018a) is another dataset (based on Zhang and Lapata (2017)) that gives multi-split simplifications. We did not adopt it here as the data came with only 359 sentences with limited variations in splitting.

[4] https://huggingface.co/facebook/bart-large

[5] https://github.com/huggingface/transformers/blob/master/examples/pytorch/translation/run_translation.py



Welcome to Text Quality Assessment IV

Introduction

The test you are about to take is part of an on-going effort to develop an AI-powered reading tool.

You find below three pieces of text, **Source**, **Text A**, and **Text B**, with A and B presented in a random order. **Source** is a text taken verbatim from Wikipedia. **Text A** and **Text B** are lightly modified versions of **Source**. Read them carefully and indicate how much you agree to statements about them, by using sliders (1 = Strongly disagree, 5 = Strongly agree) or respond to questions by clicking buttons.

Please note: Punctuations (including apostrophes) are deliberately set apart. Don't count them as errors. Leaving any of the sliders at default position (0) or radio buttons unchecked will result in an automatic rejection.

Problem Section

Source

Akeem Priestley is in the Jackson Dolphins club and he plays for the Connecticut Huskies youth team as well as for Sheikh Russel KC .

Text B

Akeem Priestley is in the Jackson Dolphins club .
Akeem Priestley plays for Sheikh Russel KC .
Akeem Priestley plays for the Connecticut Huskies youth team .

Text A

Akeem Priestley is in the Jackson Dolphins club and plays for Sheikh Russel KC .
He played for the youth club Connecticut Huskies .

Figure 1: A screen capture of HIT. This is what a Turker would be looking at when taking the test.

4 Preliminary Analysis

Table 3 summarizes results from comparison questions. A question, labelled $\langle\langle$S, BART-A$\rangle\rangle_{|q}$, asks a Turker, which of Source and BART-A he or she finds easier to understand, where BART-A is a BART generated two-sentence simplification. We had 791 (113×7) responses, out of which 32% said they preferred Source, 67% liked BART better, and 1% replied they were not sure. Another question, labelled $\langle\langle$S, HUM-A$\rangle\rangle_{|q}$, compares Source to HUM-A, a two-sentence split by human. It got 756 responses (108×7). The result is generally parallel to $\langle\langle$S, BART-A$\rangle\rangle_{|q}$. The majority of people favored a two-sentence split over a complex sentence. The fact that three sentence versions are also favored over complex sentences suggests that breaking up a complex sentence improves readability, regardless of how many pieces it ends up with.

Table 4 gives a tally of responses to comparison questions on two- and three-sentence splits. More people voted for bipartite over tripartite simplifications. Tables 5 and 6 show scores on fluency, grammar, and meaning retention of simplifications, comparing BART-A and HUM-B,[6] on one hand, and HUM-A and HUM-S, on another, on a scale of 1 (poor) to 5 (excellent). In either case, we did not see much divergence between A and B in grammar and meaning, but they diverged the most in fluency. A T-test found the divergence statistically significant. Two-sentence simplifications generally scored higher on fluency (over 4.0) than three sentence counterparts (below 4.0).

Table 7 gives an example showing what generated texts looked like in BART-A and HUM-A/B.

[6] As Tables 5 and 6 indicate, BART-A is generally comparable to HUM-A in the quality of its outputs, suggesting that what it generates is mostly indistinguishable from those by humans.

QUESTION	AVAILABLE CHOICES				
	S	BART-A	HUM-B	NOT SURE	TOTAL
$\langle\!\langle$S, BART-A$\rangle\!\rangle_{\vert q}$	254 (0.32)	527 (0.67)	–	10 (0.01)	791
$\langle\!\langle$S, HUM-B$\rangle\!\rangle_{\vert q}$	290 (0.37)	–	490 (0.62)	11 (0.01)	791
	S	HUM-A	HUM-B	NOT SURE	TOTAL
$\langle\!\langle$S, HUM-A$\rangle\!\rangle_{\vert q}$	253 (0.33)	494 (0.65)	–	9 (0.01)	756
$\langle\!\langle$S, HUM-B$\rangle\!\rangle_{\vert q}$	288 (0.38)	–	463 (0.61)	5 (0.01)	756

Table 3: Results from the Comparison Section. We are showing how many Turkers went with each available choice. S: source. BART-A: BART-generated two-sentence simplification. HUM-A: manual two-sentence simplification. HUM-B: manual three-sentence simplification. $\langle\!\langle$S, BART-A$\rangle\!\rangle_{|q}$ asked Turkers which of S and BART-A they found easier to understand. 67% said they would favor BART-A, and 32% S, with 1% not sure. $\langle\!\langle$S, HUM-B$\rangle\!\rangle_{|q}$ compares S and HUM-B for readability. $\langle\!\langle$S, HUM-A$\rangle\!\rangle_{|q}$ looks at S and HUM-A.

QUESTION	AVAILABLE CHOICES			
	BART-A	HUM-B	NOT SURE	TOTAL
$\langle\!\langle$BART-A, HUM-B$\rangle\!\rangle_{\vert q}$	460 (0.58)	316 (0.40)	15 (0.02)	791
	HUM-A	HUM-B	NOT SURE	TOTAL
$\langle\!\langle$HUM-A, HUM-B$\rangle\!\rangle_{\vert q}$	439 (0.58)	301 (0.40)	16 (0.02)	756

Table 4: Comparison of two- vs three-sentence simplifications. The majority went with two-sentence simplifications regardless of how they were generated.

category	HUM-A	HUM-B
**fluency	4.04 (0.39)	3.75 (0.38)
grammar	4.12 (0.32)	4.10 (0.32)
meaning	4.31 (0.36)	4.33 (0.28)

Table 5: Average scores and standard deviations for HUM-A and HUM-B. HUM-A is more fluent than HUM-B. Note: ** = $p < 0.01$.

category	BART-A	HUM-B
**fluency	4.04 (0.37)	3.72 (0.36)
grammar	4.07 (0.30)	4.05 (0.34)
meaning	4.21 (0.38)	4.25 (0.35)

Table 6: Average scores and standard deviations of BART-A and the corresponding HUM-B. BART-A is significantly more fluent than HUM-B. '**' indicates the two groups are distinct at the 0.01 level.

5 A Bayesian Perspective

A question we are curious about at this point is what are the factors that led Turkers to decisions that they made. We answer the question by way of building a Bayesian model based on predictors assembled from the past literature on readability and in related fields.

5.1 Model

We consider a Bayesian logistic regression.[7]

$$\begin{aligned} Y_j &\backsim Ber(\lambda), \\ \text{logit}(\lambda) &= \beta_0 + \sum_{i}^{m} \beta_i X_i, \qquad (1) \\ \beta_i &\backsim \mathcal{N}(0, \sigma_i) \ \ (0 \leq i \leq m) \end{aligned}$$

$Ber(\lambda)$ is a Bernoulli distribution with a parameter λ. β_i represents a coefficient tied to a random variable (predictor) X_i, where β_0 is an intercept. We assume that β_i, including the intercept, follows a normal distribution with the mean at 0 and the variance at σ_i. Y_i takes either 1 or 0. $Y = 1$ if a Turker finds a two-sentence simplification more readable, and $Y = 0$ if a three-sentence version is preferred.

[7] Equally useful in explaining relationships between potential causes and the outcome are Bayesian tree-based methods (Chipman et al., 2010; Linero, 2017; Nuti et al., 2019), which we do not explore here. The latter could become a viable choice when an extensive non-linearity exists between predictors and the outcome.

[8] https://github.com/jasonyux/FastKASSIM

[9] https://github.com/luozhouyang/python-string-similarity

[10] https://github.com/shivam5992/textstat

TYPE	TEXT
ORIGINAL	The Alderney Airport serves the island of Alderney and its 1st runway is surfaced with poaceae and has a 497 meters long runway .
BART-A	Alderney Airport serves the island of Alderney . The 1st runway at Aarney Airport is surfaced with poaceae and has 497 meters long .
HUM-A	The runway length of Alderney Airport is 497.0 and the 1st runway has a poaceae surface . The Alderney Airport serves Alderney .
HUM-B	The surface of the 1st runway at Alderney airport is poaceae . Alderney Airport has a runway length of 497.0 . The Alderney Airport serves Alderney .

Table 7: Original vs. Modified

CATEGORY	VAR NAME	DESCRIPTION	VALUE
synthetic	**bart**	true if the simplification is generated by BART; false otherwise.	categorical
cohesion	**ted1**	the tree edit distance (TED) between a source and its proposed simplification.[8] where TED represents the number of editing operations (*insert*, *delete*, *replace*) required to turn one parse tree into another; the greater the number, the less the similarity (Boghrati et al., 2018; Zhang and Shasha, 1989).	continuous
	ted2	TED across sentences contained in the simplification.	continuous
	subset	Subset based Tree Kernel (Collins and Duffy, 2002; Moschitti, 2006; Chen et al., 2022)[8]	continuous
	subtree	Subtree based Tree Kernel (Collins and Duffy, 2002; Moschitti, 2006; Chen et al., 2022)[8]	continuous
	overlap	Szymkiewicz-Simpson coefficient, a normalized cardinality of an intersection of two sets of words (Vijaymeena and Kavitha, 2016).[9]	continuous
cognitive	**frazier**	the distance from a terminal to the root or the first ancestor that occurs leftmost (Frazier, 1985).	continuous
	yngve	per-token count of non-terminals that occur to the right of a word in a derivation tree (Yngve, 1960).	continuous
	dep length	per-token count of dependencies in a parse (Magerman, 1995; Roark et al., 2007).	continuous
	tnodes	per-token count of nodes in a parse tree (Roark et al., 2007)	continuous
classic	**dale**	Dale-Chall readability score (Chall and Dale, 1995)[10]	continuous
	ease	Flesch Reading Ease (Flesch, 1979)[10]	continuous
	fk grade	Flesch-Kincaid Grade Level (Kincaid et al., 1975)[10]	continuous
perception	**grammar**	grammatical integrity (manually coded)	continuous
	meaning	semantic fidelity (manually coded)	continuous
	fluency	language naturalness (manually coded)	continuous
structural	**split**	true if the sentence is bisected; false otherwise.	categorical
informational	**samsa**	measures how much of the original content is preserved in the target (Sulem et al., 2018b).	continuous

Table 8: Predictors

5.2 Predictors

We use predictors shown in Table 8. They come in six categories: *synthetic*, *cohesion*, *cognitive*, *classic*, *perception* and *structural*. A *synthetic* feature indicates whether the simplification was created with BART or not, taking *true* if it was and *false* otherwise. Those found under *cohesion* are our adaptions of SYNSTRUT and CRFCWO, which are among the diverse features McNamara et al. (2014) created to measure cohesion across sentences. SYSTRUCT gauges the uniformity and consistency across sentences by looking at their syntactic similarities, or by counting nodes in a common subgraph shared by neighboring sentences. We substituted SYSTRUCT with **tree edit distance** (Boghrati et al., 2018), as it allows us to handle multiple subgraphs, in contrast to SYSTRUCT, which only looks for a single common subgraph. CRFCWO gives a normalized count of tokens found in common between two neighboring sentences. We emulated it here with the Szymkiewicz-Simpson coefficient, given as $O(X,Y) = \frac{|X \cap Y|}{\min(|X|,|Y|)}$.

Predictors in the *cognitive* class are taken from works in clinical and cognitive linguistics (Roark et al., 2007; Boghrati et al., 2018). They reflect various approaches to measuring the cognitive complexity of a sentence. For example, **yngve** scoring defines a cognitive demand of a word as the number of non-terminals to its right in a derivation rule that are yet to be processed.

5.2.1 yngve

Consider Figure 2. **yngve** gives every edge in the parse a number reflecting its cognitive cost. NP gets '1' because it has a sister node VP to its right. The cognitive cost of a word is defined as the sum of numbers on a path from the root to the word. In Figure 2, 'Vanya' would get $1 + 0 + 0 = 1$, whereas 'home' 0. Averaging words' costs gives us an Yngve complexity.

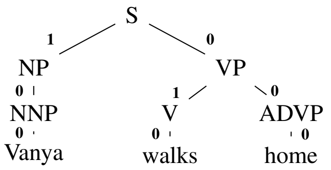


Figure 2: Yngve scoring

5.2.2 frazier

frazier scoring views the syntactic depth of a word (the distance from a leaf to a first ancestor that occurs leftmost in a derivation rule) as a most important factor to determining the sentence complexity. If we run **frazier** on the sentence in Figure 2, it will get the score like one shown in Figure 3. 'Vanya' gets $1 + 1.5 = 2.5$, 'walks' 1 and 'home' 0 (which has no leftmost ancestor). Roark et al. (2007) reported that both **yngve** and **frazier** worked well in discriminating subjects with mild memory impairement.

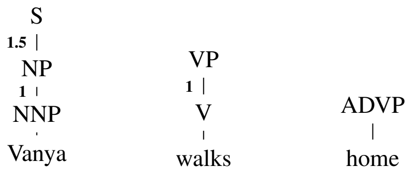


Figure 3: Frazier scoring

5.2.3 dep length

dep length (dependency length) and **tnodes** (tree nodes) are also among the features that Roark et al. (2007) found effective. The former measures the number of dependencies in a dependency parse, and the latter the number of nodes in a phrase structure tree.

5.2.4 subset and subtree

subset and **subtree** are both measures based on the idea of *Tree Kernel* (Collins and Duffy, 2002; Moschitti, 2006; Chen et al., 2022).[11] The former considers how many subgraphs two parses share, while the latter how many subtrees. Note that subtrees are those structures that end with terminal nodes.

5.2.5 Classic readability features

We also included features that have long been established in the readability literature as standard, i.e. Dale-Chall Readability, Flesch Reading Ease, and Flesch-Kincaid Grade Level (Chall and Dale, 1995; Flesch, 1979; Kincaid et al., 1975).

[11] Tree Kernel is a function defined as $K(T_1, T_2) = \sum_{n_1 \in N(T_1)} \sum_{n_2 \in N(T_2)} \Delta(n_1, n_2)$ where

$$\Delta(a,b) = \begin{cases} 0 & \text{if } a \neq b; \\ 1 & \text{if } a = b; \\ \prod_i^{C(a)} (\sigma + \Delta(c_a^{(i)}, c_b^{(i)})) & \text{otherwise.} \end{cases}$$

$C(a)$ = the number of children of a, $c_a^{(i)}$ represents the i-th child of a. We let $\sigma > 0$.

5.2.6 Perceptual features

Those found in the *perception* category are from judgments Turkers made on the quality of simplifications we asked them to evaluate. We did not provide any specific definition or instruction as to what constitutes grammaticality, meaning, and fluency during the task. So, it is most likely that their responses were spontaneous and perceptual.

5.2.7 **split** and **samsa**

Finally, we have **split**, which records whether or not the simplification is bipartite: it takes *true* if it is, and *false* if not. **samsa** is a recent addition to a battery of simplification metrics, which looks at how much of a propositional content in the source remains after a sentence is split (Sulem et al., 2018b). (The greater, the better.) We standardized all of the features, except for **bart** and **split**, by turning them into z-scores, where $z = \frac{x-\bar{x}}{\sigma}$.

5.3 Evaluation

We trained the model (Eqn. 1) using BAMBI (Capretto et al., 2020),[12] with the burn-in of 50,000 while making draws of 4,000, on 4 MCMC chains (Hamiltonian). As a way to isolate the effect (or importance) of each predictor, we did two things: one was to look at a posterior distribution of each factor, i.e. a coefficient β tied with a predictor, and see how far it is removed from 0; another was to conduct an ablation study where we looked at how the absence of a feature affected the model's performance, which we measured with a metric known as 'Watanabe-Akaike Information Criterion' (WAIC) (Watanabe, 2010; Vehtari et al., 2016), a Bayesian incarnation of AIC (Burnham and Anderson, 2003).[13]

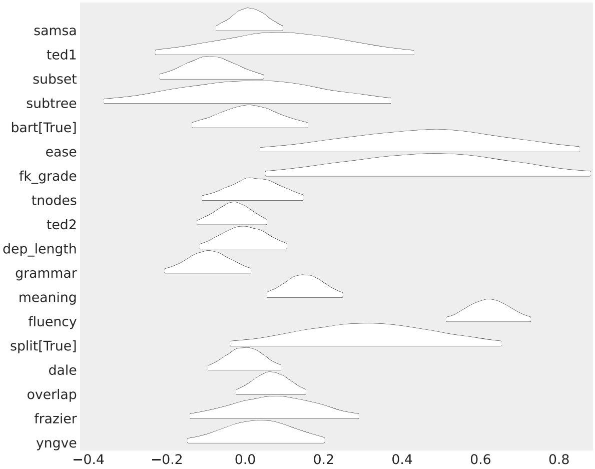


Figure 4: Posterior distributions of coefficients (β's) in the full model. The further the distribution moves away from 0, the more relevant it becomes to predicting the outcome.

Figure 4 shows what posterior distributions of parameters associated with predictors looked like after 4,000 draw iterations with MCMC. None of the chains associated with the parameters exhibited divergence. We achieved $\hat{R}$ between 1.0 and 1.02, for all β_i, a fairly solid stability (Gelman and Rubin, 1992), indicating that all the relevant parameters had successfully converged.[14]

At a first glance, it is a bit challenging what to make of Figure 4, but a generally accepted rule of thumb is to assume distributions that center around 0 as of less importance in terms of explaining observations, than those that appear away from zero. If we go along with the rule, then the most likely candidates that affected readability are: **ease**, **subset**, **fk grade**, **grammar**, **meaning**, **fluency**, **split**, and **overlap**. What remains unclear is, to what degree the predictors affected readability.

One good way to find out is to do an ablation study, a method to isolate the effects of an individual factor by examining how seriously its removal from a model degrades its performance. The result of the study is shown in Table 9. Each row represents performance in WAIC of a model with a particular predictor removed. Thus, 'ted1' in Table 9 represents a model that includes all the predictors in Table 8, except for **ted1**. A row in blue represents a full model which had none of the features disabled. Appearing above the base model means that a removal of a feature had a positive effect, i.e. the feature is redundant. Appearing below means that the removal had a negative effect, indicating that we should not forgo the feature. A

[12] `https://bambinos.github.io/bambi/main/index.html`

[13] WAIC is given as follows.

$$\text{WAIC} = \sum_{i}^{n} \log \mathbb{E}[p(y_i|\theta)] - \sum_{i}^{n} \mathbb{V}[\log p(y_i|\theta)]. \quad (2)$$

$\mathbb{E}[p(y_i|\theta)]$ represents the average likelihood under the posterior distribution of θ, and $\mathbb{V}[\alpha]$ represents the sample variance of α, i.e. $\mathbb{V}[\alpha] = \frac{1}{S-1}\sum_{1}^{S}(\alpha_s - \bar{\alpha})$, where α_s is a sample draw from $p(\alpha)$. A higher WAIC score indicates a better model. n is the number of data points.

[14] $\hat{R}$ = the ratio of within- and between-chain variances, a standard tool to check for convergence (Lambert, 2018). The closer the ratio is to the unity, the more likely MCMC chains have converged.

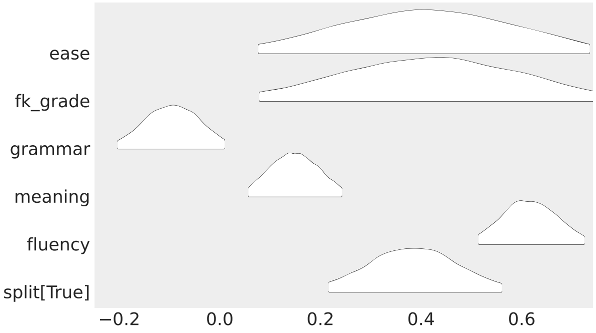


Figure 5: Posterior distributions of the coefficient parameters in the reduced model.

feature becomes more relevant as we go down, and becomes less relevant as we go up the table. Thus the most relevant is **fluency**, followed by **meaning**, the least relevant is **subtree**, followed by **dale**, and so forth. We can tell from Table 9 what predictors we need to keep to explain the readability: they are **grammar**, **split**, **fk grade**, **ease**, **meaning** and **fluency** (call them 'select features'). Note that **bart** is in the negative realm, meaning that from a perspective of readability, people did not care about whether the simplification was done by human or machine. **samsa** was also found in the negative domain, implying that for a perspective of information, a two-sentence splitting carries just as much information as a three way division of a sentence.

To further nail down to what extent they are important, we ran another ablation experiment involving the select features alone. The result is shown in Table 10. At the bottom is **fluency**, the second to the bottom is **split**, followed by **meaning**, and so forth. As we go up the table, a feature becomes less and less important. The posterior distributions of these features are shown in Figure 5.[15] Not surprisingly, they are found away from zero, with **fluency** furtherest away. The result indicates that contrary to the popular wisdom that classic readability metrics such as **ease**, and **fk grade**, are of little use, they had a large sway on decisions people made when they were asked about readability.

6 Conclusions

In this work, we asked two questions: does cutting up a sentence help the reader better understand the text? and if so, does it matter how many pieces we break it into? We found that splitting does allow the reader to better interact with the text (Table 3) and moreover, two-sentence simplifications are clearly favored over three-sentence simplifications (Tables 3,9,10). Why two-sentence splits make a better simplification is something of a mystery. A possible answer may lie in a potential disruption splitting may have caused in a sentence-level discourse structure, whose integrity Crossley et al. (2011, 2014) argued, constitutes a critical part of simplification, a topic that we believe is worth a further exploration in the future.

7 Limitations

- We did not consider cases where a sentence is split into more than three. This is mainly due to our failure to find a dataset containing manual simplifications of length greater than three in a large number. While it is unlikely that our claim in this work does not hold for cases beyond three, testing the hypothesis on cases that involve more than three sentences would be desirable.

- A cohort of people we solicited for the current work are generally well educated adults who speak English as the first language. Therefore, the results we found in this work may not necessarily hold for L2-learners, minors, or those who do not have college level education.

8 Acknowledgement

We thank anonymous reviewers for sharing with us their comments and ideas. We note their effort with much gratitude and appreciation.

References

Reihane Boghrati, Joe Hoover, Kate M. Johnson, Justin Garten, and Morteza Dehghani. 2018. Conversation level syntax similarity metric. *Behavior Research Methods*, 50(3):1055–1073.

Jan A. Botha, Manaal Faruqui, John Alex, Jason Baldridge, and Dipanjan Das. 2018. Learning to split and rephrase from Wikipedia edit history. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 732–737, Brussels, Belgium. Association for Computational Linguistics.

K.P. Burnham and D.R. Anderson. 2003. *Model Selection and Multimodel Inference: A Practical*

[15] We found that they had $1.0 \leq \hat{R} \leq 1.01$, a near-perfect stability. Settings for MCMC, i.e. the number of burn-ins and that of draws, were set to the same as before.

effect	predictor	rank↑	waic↑	p_waic↓	d_waic↓	se↓	dse↓
–	subtree	0	-1899.249	17.797	0.000	17.787	0.000
	dale	1	-1899.287	17.852	0.038	17.791	0.207
	dep_length	2	-1899.362	17.916	0.113	17.777	0.211
	yngve	3	-1899.406	17.904	0.157	17.777	0.464
	tnodes	4	-1899.414	17.898	0.165	17.797	0.408
	bart	5	-1899.421	17.967	0.172	17.786	0.216
	samsa	6	-1899.450	18.018	0.201	17.776	0.315
	ted1	7	-1899.557	17.996	0.308	17.771	0.575
	ted2	8	-1899.632	18.019	0.383	17.782	0.624
	frazier	9	-1899.740	18.096	0.492	17.779	0.708
	subset	10	-1900.069	17.811	0.820	17.741	1.282
	overlap	11	-1900.431	17.966	1.182	17.750	1.511
ref.	base	12	-1900.532	19.089	1.283	17.787	0.208
+	grammar	13	-1900.780	17.979	1.531	17.698	1.657
	split	14	-1900.852	18.030	1.603	17.697	1.776
	ease	15	-1901.657	17.962	2.408	17.670	2.064
	fk_grade	16	-1901.710	18.030	2.462	17.685	2.049
	meaning	17	-1903.795	17.885	4.546	17.425	3.071
	fluency	18	-1965.386	17.938	66.137	14.067	11.349

Table 9: Comparison in WAIC. *p_waic* = the effective number of parameters (Spiegelhalter et al., 2002), a measure to estimate the complexity of the model: the greater, the more complex. *d_waic* = the distance in WAIC to the top model. *se* = standard error of WAIC estimates. *dse* = standard error of differences in WAIC estimates between the top model and each of the rest. ↑ means that higher is better. ↓ indicates the opposite.

predictor	rank↑	waic↑	p_waic↓	d_waic↓	se↓	dse↓
base	0	-1891.901	7.181	0.000	17.485	0.000
grammar	1	-1892.235	6.183	0.335	17.365	1.672
ease	2	-1893.515	6.137	1.614	17.350	2.324
fk_grade	3	-1893.626	6.161	1.726	17.366	2.358
meaning	4	-1895.308	6.145	3.407	17.111	3.059
split	5	-1900.028	6.169	8.127	17.038	4.247
fluency	6	-1956.041	5.935	64.140	13.784	11.289

Table 10: Results in WAIC for the reduced model

Information-Theoretic Approach. Springer New York.

Tomás Capretto, Camen Piho, Ravin Kumar, Jacob Westfall, Tal Yarkoni, and Osvaldo A. Martin. 2020. Bambi: A simple interface for fitting bayesian linear models in python.

J.S. Chall and E. Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula.* Brookline Books.

Maximillian Chen, Caitlyn Chen, Xiao Yu, and Zhou Yu. 2022. Fastkassim: A fast tree kernel-based syntactic similarity metric. *arXiv preprint arXiv:2203.08299.*

Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. 2010. BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1).

Michael Collins and Nigel Duffy. 2002. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 263–270, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Scott A. Crossley, David B. Allen, and Danielle S. McNamara. 2011. Text readability and intuitive simplification: A comparison of readability formulas. *Reading in a Foreign Language*, 23(1):84–101.

Scott A. Crossley, Hae Sung Yang, and Danielle S. McNamara. 2014. What's so simple about simplified texts? A computational and psycholinguistic investigation for text comprehension and text processing. *Reading in a Foreign Language*, 26(1):92–113.

R. Flesch. 1979. *How to Write Plain English: A Book for Lawyers and Consumers.* Harper & Row.

Lyn Frazier. 1985. Syntactic complexity. In David R. Dowty, Lauri Karttunen, and Arnold M.Editors Zwicky, editors, *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives*, Studies in Natural Language Processing, pages 129–189. Cambridge University Press.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada. Association for Computational Linguistics.

Andrew Gelman and Donald B. Rubin. 1992. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472.

Joongwon Kim, Mounica Maddela, Reno Kriz, Wei Xu, and Chris Callison-Burch. 2021. BiSECT: Learning to split and rephrase sentences with bitexts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6193–6209, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

J. Peter Kincaid, Robert P. Fishburne Jr., Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, Naval Technical Training Command.

Ben Lambert. 2018. *A Student's Guide to Bayesian Statistics.* SAGE.

Antonio R. Linero. 2017. A review of tree-based bayesian methods. *Communications for Statistical Applications and Methods*, 24:543–559.

David M. Magerman. 1995. Statistical decision-tree models for parsing. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 276–283, Cambridge, Massachusetts, USA. Association for Computational Linguistics.

Danielle S. McNamara, Arthur C. Graesser, Philip M. McCarthy, and Zhiqiang Cai. 2014. *Automated Evaluation of Text and Discourse with Coh-Metrix.* Cambridge University Press.

Alessandro Moschitti. 2006. Making tree kernels practical for natural language learning. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 113–120, Trento, Italy. Association for Computational Linguistics.

Shashi Narayan, Claire Gardent, Shay B. Cohen, and Anastasia Shimorina. 2017. Split and rephrase. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 606–616, Copenhagen, Denmark. Association for Computational Linguistics.

Christina Niklaus, André Freitas, and Siegfried Handschuh. 2019. MinWikiSplit: A sentence splitting corpus with minimal propositions. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 118–123, Tokyo, Japan. Association for Computational Linguistics.

Giuseppe Nuti, Lluís Antoni Jiménez Rugama, and Andreea-Ingrid Cross. 2019. An explanable bayesian decision tree algorithm. *arXiv:1901.03214v3 [stat.ML].*

Brian Roark, Margaret Mitchell, and Kristy Hollingshead. 2007. Syntactic complexity measures for detecting mild cognitive impairment. In *Biological, translational, and clinical language processing*, pages 1–8, Prague, Czech Republic. Association for Computational Linguistics.

Steven Ross, Michael H. Long, and Yasukata Yano. 1991. Simplification or elaboration? the effects of two types of text modifications on foreign language reading comprehension. *University of Hawai'i Working Papers in ESL*, 10(22):1–32.

David J Spiegelhalter, Nicola G Best, Bradley P Carlin, and Angelika Van Der Linde. 2002. Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):583–639.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018a. BLEU is not suitable for the evaluation of text simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 738–744. Association for Computational Linguistics.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018b. 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.

Aki Vehtari, Andrew Gelman, and Jonah Gabry. 2016. Practical bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432.

M. K. Vijaymeena and K. Kavitha. 2016. A survey on similarity measures in text mining. *Machine Learning and Applications: An International Journal*, 3(1):19–28.

Sumio Watanabe. 2010. Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, pages 3571–3594.

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*, 3:283–297.

Victor H. Yngve. 1960. A model and an hypothesis for language structure. *Proceedings of the American Philosophical Society*, 104(5):444–466.

Kaizhong Zhang and Dennis Shasha. 1989. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal on Computing*, 18(6):1245–1262.

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.