CEScore: Simple and Efficient Confidence Estimation Model for Evaluating Text Simplification

Anonymous ACL submission

Abstract

Evaluating quality of text simplification (TS) is challenging, especially for models following unsupervised or reinforcement learning techniques, where reference data is unavailable. We introduce CEScore, a novel statistical model that evaluates TS quality without relying on references. By mimicking the way humans evaluate TS, CEScore provides 4 metrics (S_{score}, G_{score}, M_{score}, and CE_{score}) to assess simplicity, grammaticality, meaning preservation, and overall quality, respectively. In an experiment with 26 TS models, CEScore correlates strongly with human evaluations, achieving 0.98 in Spearman correlations at model-level. This underscores the potential of CEScore as a simple and efficient metric for assessing the quality of TS models.

1 Introduction

The exponential growth of digital content in recent years has intensified the demand for precise and efficient text simplification (TS) models. These models are designed to alter text, enhancing its comprehensibility while preserving its fundamental essence and the majority of its original meaning (Siddharthan, 2002). This intricate process rendering the text accessible to a broader audience, including non-native speakers, children, and those grappling with conditions. Furthermore, TS plays a pivotal role as a preprocessing step in natural language processing (NLP) tasks (AlAjlouni and Li, 2023). However, improving TS models hinges on our capability to accurately and efficiently evaluate the quality of their outputs.

The absence of dedicated automatic evaluation metrics for TS has resulted in the adaptation of metrics originally designed for assessing Machine Translation (MT). This approach arises from the recognition that TS essentially involves monolingual translation, where complex text is transformed into simpler language within the same language. Prominent evaluation metrics such as BLEU (Papineni et al., 2002) and BERT score (Zhang et al., 2019) have been repurposed for this task. However, based on previous researches (Xu et al., 2016; Sulem et al., 2018a,b), the use of these metrics has not proven entirely effective in evaluating TS. This inadequacy stems from a fundamental issue: these metrics tend to favor longer sentences, which directly contradicts the essence of TS, where the goal is to make text simpler with shorter sentences. Consequently, the quality of TS model evaluation has suffered, prompting a growing need for specialized evaluation metrics that better align with the unique goals and challenges of text simplification.

This paper presents a novel statistical model called CEScore, which measures the quality of TS. Unlike traditional evaluation methods that rely on reference texts, CEScore directly evaluates the quality of a simplified text by considering how well it adheres to three fundamental dimensions: Simplicity (S), ensuring that the text becomes more straightforward; Meaning preservation (M), verifying that the essence of the original content remains intact; and grammaticality (G), assessing the text's adherence to proper grammar.

CEScore generates four distinct scores: S_{score}, G_{score}, M_{score}, and CE_{score}, each of which represents the model's assessment for specific criteria within the simplification process. This approach mirrors the way humans naturally evaluate TS, providing a more contextually relevant and interpretable assessment of the quality of a simplified text, thereby dispensing with the need for reference sentences.

In our quest to compute S_{score}, we have developed new statistical formulas for evaluating sentence simplicity. These formulas, known as the *Sentence Length Score* (SLS), *Average Sentence Familiarity* (ASF), and *Text Simplicity Score* (TSS), designed to evaluate text simplicity based on many factors influence the simplicity of a text, such as

word count, clause structure, word count within each clause, and the familiarity of vocabulary used.

To test CEScore's effectiveness, we compared its scores with others metrics that commonly used for evaluating TS systems, including BLEU, SARI (Xu et al., 2016), BERTscore, and SAMSA(Sulem et al., 2018b). Our comparison utilizes a sizable human evaluation benchmark provided by Sulem et al. (2018c) as a foundational reference for this comparison. We performed that comparison of two levels: sentence-level and model-level.

2 Related Work

2.1 Manual evaluation Method

The absence of a robust automated evaluation metric for TS has led researchers to rely on manual evaluation methods (Alva-Manchego et al., 2020), where human evaluators carefully scrutinize the outcomes generated by TS models. Typically, the evaluators assess the model's outputs based on fundamental criteria such as, grammaticality (G), Meaning Preservation (M), Simplicity (S).

Annotators assign scores to each criterion individually, G and M are assessed on a scale of 1 to 5, while S is evaluated on a scale of -2 to 2, wherein zero signifies an equivalent degree of simplicity (Narayan and Gardent, 2014; Nisioi et al., 2017; Narayan et al., 2017; Sulem et al., 2018c; Niklaus et al., 2019).

While manual evaluation remains the gold standard for evaluating TS models, it is not without its limitations. This approach is inherently time-consuming and requires significant human labor resources (Papineni et al., 2002). Furthermore, the inherent subjectivity in manual evaluation introduces variability into the results, stemming from the diverse backgrounds and biases of the evaluators (Xu et al., 2016).

2.2 Automatic Evaluation Method

In the realm of automatic evaluation metrics for TS, two distinct approaches have emerged. The first approach relies on references, wherein the output of a TS model is evaluated by comparing it to reference texts or human-generated simplifications. Metrics like BLEU, SARI, and BERTscore fall under this category.

BLEU is an n-gram-based evaluation metric widely used for assessing MT quality. It has been employed as an automatic metric for evaluating TS models (Narayan et al., 2017; Aharoni and Goldberg, 2018; Botha et al., 2018; Niklaus et al., 2019). While BLEU exhibits a strong correlation with human judgments regarding G and M, studies have shown its limitations in predicting S, whether in terms of lexical simplification (Xu et al., 2016) or structural simplification (Sulem et al., 2018a).

In response to BLEU's limitations, Xu et al. (2016) introduced SARI as a reference metric specifically designed for the evaluation of TS models that emphasize lexical simplification. It compares the n-grams of the output with those in the input and human-generated references. However, Sulem et al. (2018a) found that neither BLEU nor SARI are well-suited for assessing TS models that involve structural simplification. Consequences, they introduced SAMSA as metric primarily tailored for the evaluation of structural TS.

BERTscore introduced by Zhang et al. in 2019, for evaluating the quality of machine-generated text. It is specifically designed to measure the similarity between the generated text and a reference text using contextual embeddings from BERT (Kenton and Toutanova., 2019). Recently, BERTScore has gained popularity as an evaluation metric for TS tasks, as it is well-suited for assessing TS models that involve both of structural and lexical TS (AlAjlouni et al., 2023).

The second approach, on the other hand, does not necessitate the presence of reference texts. These metrics are often referred to as Confidence Estimation (CE) or Quality Estimation (QE) (Blatz et al., 2004). They evaluate the quality of simplified text solely based on the input text and the output produced by the TS model, without the need for reference comparisons. This approach is particularly useful when reference simplifications are scarce or unavailable, providing a more versatile and practical means of assessing the performance of TS models. This concept initially emerged within the domain of MT for evaluating the quality of automatically translated text (Blatz et al., 2004; Specia et al., 2009; Martins et al., 2017; Specia et al., 2018).

Sulem et al. in 2018 introduced SAMSA as first reference-less automatic metric to address TS. It employs semantic parsing to assess the quality of simplification by breaking down the input text based on its scenes and comparing it to the output. SAMSA penalizes cases where the number of sentences in the output is more than the number of scenes in the input. SAMSA_{abl} is a modified

version of SAMSA in which the penalization condition is omitted. However , SAMSA relies on the TUPA parser (Hershcovich et al., 2017) to decompose source sentences into their constituent scenes. This dependency has limited the practicality of SAMSA, primarily due to the parser's accuracy is often compromised, given the complexity of the language, which has adversely affected the accuracy of SAMSA. As a solution, the authors resorted to manually decomposing source sentences, but this adaptation eliminates SAMSA advantageous as a reference-less automatic metric.

3 CEScore Model

The CEScore model, where CEScore stands for *Confidence Estimation Score*, represents a statistical model tailored to assess the quality of TS without requiring reference texts for comparison. This model emulates the human approach. It generates four distinct metrics: S_{score}, G_{score}, M_{score}, and CE_{score}, each of which represents the model's assessment for specific criteria. The CEScore Model takes two input texts: the complex text (T_C) and the simplified text (T_S). It computes the S_{score}, G_{score} and M_{score} by calling the SScore, MScore and GScore functions, respectively. The CE_{score} is calculated by taking the geometric mean of the three scores: S_{score}, G_{score} and M_{score}.

3.1 SScore Function

The SScore is a statistical function designed to estimate S criterion. There are many factors influence the simplicity of a text, such as word count, clause structure, word count within each clause, and the familiarity of vocabulary used.

The *Sentence Length Score* (SLS) as shown in Equation 1 is a novel formula developed to normalize the sentence length.

$$SLS(S) = 1 - \frac{1}{1 + e^{\tau(|S_{tokens}| - \omega)}} \qquad (1)$$

In this equation, S_{tokens} is a list of tokens (words) that are belong to the sentence S after removing non-alphabetic and non-numerical tokens. The constants τ and ω are used to control the SLS range, which are set to 0.22 and 13 for τ and ω, respectively.

The *Average Sentence Familiarity* (ASF) is a novel formula we have developed to measure the familiarity of a sentence for a broad audience. To accomplish this, we have incorporated two key scales: the *percent_known* scale from Brysbaert's concreteness list (Brysbaert et al., 2014) and the $Zipf$ scale from the *SUBTLEX-US* frequency list (Heuven et al., 2014).

In Brysbaert's concreteness list, the *percent_known* value for a word signifies the percentage of participants who recognized that word. Conversely, the $Zipf$ scale is a frequency measure extracted from the *SUBTLEX-US* frequency list. This scale offers a more accessible way to comprehend word frequency compared to traditional measures. $Zipf$ values range from 1 to 7, with values 1-3 denoting low-frequency words and values 4-7 indicating high-frequency words. Table 1 presents the algorithm that we follow in the ASF function.

The *Text Simplicity Score* (TSS) is a novel formula devised to measure the simplicity of a given text. As outlined in Table 2, the $\mathcal{F}_{strc}$ score assesses the sentence considered the most complex among all sentences within the set $\mathbb{S}$. The $\mathcal{TSS}_{score}$ provides an estimation of the text's simplicity by considering both lexical and structural simplification aspects. The SScore function employs the TSS formula to evaluate the simplicity

Algorithm 2: ASF Formula	
1:	**Function** $ASF(S)$
2:	$\mathcal{S}_{tokens} = \{s \mid s \in Split(S)\}$
3:	$\psi = \{w \mid w \in \textit{SUBTLEX-US}\}$
4:	$\check{\mathcal{S}} = \mathcal{S}_{tokens} \cap \psi$
5:	$\mathcal{ASF}_{score} = \frac{1}{\lvert\check{\mathcal{S}}\rvert} \sum_{t \in \check{\mathcal{S}}} \frac{\aleph(t) \times \hbar(t)}{\Im(t)}$
6:	**Return** $\mathcal{ASF}_{score}$

Table 1: shows the steps that the ASF formula performs to calculate $\mathcal{ASF}_{score}$

Algorithm 3: TSS Formula	
1:	**Function** $TSS(T)$
2:	$\mathbb{S} = \{s \mid s \in ToSentences(T)\}$
3:	$\mathcal{F}_{lexl} = ASF(T) + 5\sqrt[3]{SLS(T)}$
4:	$\mathcal{F}_{strc} = \min_{s \in \mathbb{S}} ASF(s) \times SLS(s)$
5:	$\mathcal{TSS}_{score} = \alpha \times \mathcal{F}_{lexl} + \beta \times \mathcal{F}_{strc}$
6:	**Return** $\mathcal{TSS}_{score}$

Table 2: shows the steps that the TSS formula performs to calculate $\mathcal{TSS}_{score}$

Algorithm 4: SScore Function	
1:	**Function** $SScore(T_C, T_S)$
2:	$S_{tss} \leftarrow TSS(T_S)$
3:	$C_{tss} \leftarrow TSS(T_C)$
4:	$S_{score} = \frac{S_{tss} - C_{tss}}{S_{tss} + C_{tss}} + 0.5$
5:	**Return** S_{score}

Table 3: shows the steps that the SScore function performs to calculate S_{score}

Algorithm 5: MScore Function	
1:	**Function** $MScore(T_C, T_S)$
2:	$\mathcal{C} = \{c \mid c \in Split(T_C)\}$
3:	$\mathcal{S} = \{s \mid s \in Split(T_S)\}$
4:	$\mathcal{I} = \mathcal{C} \cap \mathcal{S}$
5:	$\mathcal{U} = \mathcal{C} \cup \mathcal{S}$
6:	$M_{score} = \frac{\sum_{t \in \mathcal{I}} \frac{1}{(1+\hbar(t))}}{\sum_{t \in \mathcal{U}} \frac{1}{(1+\hbar(t))}}$
7:	**Return** M_{score}

Table 4: shows the steps that the MScore function performs to calculate M_{score}

of T_C and T_S, as shown in Table 3. The S_{score} is determined as the relative difference between the TSS scores of the simplified and complex texts, normalized within the range [0, 1]

3.2 MScore Function

The MScore function taking two text inputs, a complex text (T_C) and a simplified text (T_S), and returning a single score (M_{score}) that signifies how faithfully T_S retains the original meaning of T_C. The M_{score} ranges from 0 to 1, a higher value suggests a better preservation of the original meaning, while a lower score may indicate a reduction in meaning.

To calculate the M_{score} we have introduced a novel approach that involves counting the common words between T_C and T_S. Recognizing that not all words contribute equally to preserving meaning, our approach includes a technique to assess the significance of each word based on the concept of entropy, which quantifies the information carried by words. Based on entropy, common words (high frequency, e.g., stop words) are considered to convey less information compared to less common words (low frequency, e.g., names of places and people).

Table 4 outlines the steps to calculate the M_{score} value,

3.3 GScore Function

The evaluation of the G criterion poses a formidable challenge, often considered one of intricate tasks in NLP. To address this inherent difficulty, we adopted an approach that leverages the grammatical structure of T_C as a reference point to estimate the G criterion of T_S, as we assume that T_C is grammatically correct and use it as a benchmark for evaluating T_S .

The proposed approach involves breaking down T_S into n-gram set and measuring the precision of matching between this set and the corresponding n-gram set from T_C. The longer n-gram matches yield more accurate estimations of grammaticality (Papineni et al., 2002). However, in the TS task, which may involves lexical simplification that primarily focus on substituting intricate vocabulary with simpler alternatives, directly identifying lengthy n-gram matches between T_S and T_C becomes more intricate. Addressing this challenge led us to devise *SemiMatch*, a novel algorithm designed to measure the semi-match between n-grams. This algorithm considers both full matches, where an entire n-gram from T_S aligns with a corresponding n-gram in T_C, and partial matches, where $(n-1)$ tokens in an n-gram from T_S correspond with $(n-1)$ tokens in an n-gram from T_C.

The objective of the *SemiMatch* function is to evaluate the presence of sequences of n tokens (n-grams) from a candidate text in a reference text. In Table 5, we outline the steps of the *SemiMatch* function

The function calculates $\mathcal{M}_{semi}$, representing the precision of n-grams from S that semi-match C, by using the ∂ function presented in Equation 2.

$$\partial(A^n, B^n) = \begin{cases} 0, & |A^n \sqcap B^n| < n-1 \\ 1, & |A^n \sqcap B^n| < n \\ \frac{n-2}{n}, & |A^n \sqcap B^n| = n-1 \end{cases} \quad (2)$$

In this equation, A^n and B^n are n-grams (tuples), each consisting of n elements, typically words or phrases. $\sqcap$ represents the intersection of two tuples, considering the order of elements. The GScore leverages the *SemiMatch* function to measure how well the grammatical structure of T_S aligns with T_C

Algorithm 6: SemiMatch Function	
1:	**Function** $SemiMatch(C, S, n)$
2:	$Cgram^n = \{cg^n \| cg^n \in$ *n-grams(C)*$\}$
3:	$Sgram^n = \{sg^n \| sg^n \in$ *n-grams(S)*$\}$
4:	$\mathcal{M}_{semi} = \frac{\sum_{i=1} \sum_{j=1} \partial(cg_i^n, sg_j^n)}{\|Sgram^n\|}$
5:	**Return** $\mathcal{M}_{semi}$

Table 5: shows the steps that the *SemiMatch* function performs to calculate $\mathcal{M}_{semi}$

Algorithm 7: GScore Function	
1:	**Function** $GScore(T_C, T_S)$
2:	$\mathbb{S} = \{s \| s \in ToSentences(T_S)\}$
3:	$\mathbb{S}M = \{\frac{1}{4} \sum_{n=4}^{7} SemiMatch(T_C, s, n) \| s \in \mathbb{S}\}$
4:	$G_{score} = \min_{sm \in \mathbb{S}M} \{sm \| sm > 0\}$
5:	**Return** G_{score}

Table 6: shows the steps that the GScore function performs to calculate G_{score}

as a reference. As outlined in Table 7, the GScore function initializes a set $\mathbb{S}$ to store individual sentences obtained from T_S. It calculates a set $\mathbb{S}M$ that contains average semi-match ($\mathcal{M}_{semi}$) scores for each sentence s in $\mathbb{S}$ based on n-grams with n ranging from 4 to 7.

The GScore function calculates the G_{score} by finding the minimum value from the $\mathbb{S}M$ set but only considering values greater than zero[1] *(line 5)*.

4 Experimental Setup

In this experiment, we assess the accuracy of the CEScore model against conventional metrics currently employed for evaluating TS models. We examine three widely used reference-based automatic metrics: BLEU, SARI, and BERTscore, as well as two reference-less automatic metrics designed for structural TS: SAMSA and its variant, SAMSA$_{abl}$.

We compare the scores from the metrics to the human evaluation scores from a sizable benchmark provided by Sulem et al. (2018c)[2] . This benchmark encompasses human judgments regarding the performance of 26 TS models in simplifying the first 70 sentences of the TurkCorpus test set (Xu et al., 2016). These TS models cover a wide range of TS transformations, covering both lexical and structural simplification.

In total, our study involved the human evaluation of 1820 sentences from 26 TS models. Each of these sentences evaluated by three native English annotators based on four criteria: Meaning preservation (M), Grammaticality (G), Simplicity (S), and Structural Simplicity (StS). The G and M criteria were rated on a scale ranging from 1 to 5, while S and StS criteria were evaluated on a scale spanning from -2 to 2.

For the reference-based automatic metrics, we utilized eight references from two sources. The first source is the original references of the TurkCorpus testset[3] (Xu et al., 2016), consisting of eight different versions of human-generated simplifications tuned for lexical TS. The second source is the HSplit corpus [4] (Sulem et al., 2018a), which comprises four different versions of human-generated simplifications tuned for structural TS. To ensure a balanced set of references in this experiment, we selected four references from each source.

We conducted a comprehensive comparative analysis between CEScore and the considered automatic metrics across four criteria: S, G, M, and overall quality. For each of these criteria, we compared the scores obtained from the automatic metrics with the human judgment scores corresponding to that specific criterion. We used EASSE tool (Alva-Manchego et al., 2019) to compute BLEU, SARI, SAMSA, and SAMSA$_{abl}$.

In the case of the overall quality criterion, we determined the overall quality using the approach introduced by AlAjlouni et al. (2023). This approach utilizes their formula known as *'Standard A'* which was found to be more accurate than the traditional arithmetic mean in combining the G, M, and S criteria to reflect the overall quality of TS. This choice was made based on their observations of correlations in human evaluations, particularly between G and S, as well as between G and M. To provide a comprehensive view, we also reported the overall quality based on the arithmetic mean.

We conduct this comparative analysis on two levels. First, at the sentence-level, we aim to assess the accuracy of CEScore in evaluating the quality

[1] We excluded scores that equal zero because they often result from sentences that lack words, typically due to a punctuation error.

[2] https://github.com/eliorsulem/simplification-acl2018

[3] https://github.com/cocoxu/simplification

[4] https://github.com/eliorsulem/HSplit-corpus

of simplification for individual sentences. We compare the scores generated by CEScore and other automatic metrics with the corresponding human evaluations for each sentence, considering all models. In essence, for each evaluation criterion, we compare the outcomes of the automatic metrics for 1,820 sentences with their corresponding human assessments.

Second, at the model (system) level, we aim to gauge the effectiveness of the CEScore in evaluating the models as a whole. This level of analysis focuses on the average ratings for each model. For each criterion, we compare the corpus-scores provided by CEScore and other automatic metrics for each of 26 models with the average human scores for each model in each criterion.

5 Results

For the S criterion, the results showed that S_{score} displayed strong correlations with human judgment scores at both sentence and model levels, as shown in table 7. At the sentence level, the correlation coefficients were 0.64 and 0.56 for Pearson and Spearman, respectively. Remarkably, at the model level, S_{score} exhibited exceptionally strong correlations, with Pearson's and Spearman's coefficients scoring at 0.86 and 0.83, respectively. This strong positive correlation is a noteworthy finding, especially when compared to other automatic metrics, both reference-based and reference-less.

Surprisingly, automatic metrics designed to be positively correlated with simplicity, such as SARI, SAMSA, and SAMSA$_{abl}$, displayed an unexpected negative correlation with human judgment under the S criterion. This surprising result may be attributed to the fact that SARI is primarily designed to estimate lexical TS only, while SAMSA and SAMSA$_{abl}$ are tailored for assessing structural TS only. In contrast, the models in our experiment cover both lexical and structural TS, which could explain the discrepancies in correlation. Figure 1 displays scatter plots that depict the relationship at the sentence-level between human judgment scores for S criterion and the corresponding scores from automatic metrics. Each data point in these plots corresponds to an individual sentence, and regression lines are included for reference.

In the case of G criterion, G_{score} outperformed the other automatic metrics with notable margin at both sentence and model levels, as shown in table 8. At model-level, G_{score} demonstrated very strong

Metric	*Sentence-Level*		*Model-Level*	
	$\mathbf{P}_\rho$	$\mathbf{S}_\rho$	$\mathbf{P}_\rho$	$\mathbf{S}_\rho$
BLEU	-0.37	-0.26	-0.69	-0.69
SARI	-0.29	-0.27	-0.8	-0.78
BERTscore	-0.44	-0.33	-0.78	-0.73
SAMSA	-0.27	-0.27	-0.72	-0.68
SAMSA$_{abl}$	-0.32	-0.33	-0.51 (0.01)	-0.47 (0.01)
$\mathbf{S}_{score}$	**0.64**	**0.56**	**0.86**	**0.83**

Table 7: The Pearson's ($\mathbf{P}_\rho$) and Spearman's ($\mathbf{S}_\rho$) coefficients, and their associated p-values (if greater than 0.005) for the correlations between the automatic metrics and human judgment under the S criterion.

Metric	*Sentence-Level*		*Model-Level*	
	$\mathbf{P}_\rho$	$\mathbf{S}_\rho$	$\mathbf{P}_\rho$	$\mathbf{S}_\rho$
BLEU	0.41	0.47	0.64	0.59
SARI	0.01	-0.08	0.09	-0.15 (0.45)
BERTscore	0.39	0.44	0.52	0.55
SAMSA	0.20	0.20	-0.04(0.86)	-0.01(0.96)
SAMSA$_{abl}$	0.34	0.33	0.58	0.50 (0.01)
$\mathbf{G}_{score}$	**0.55**	**0.53**	**0.89**	**0.85**

Table 8: The Pearson's ($\mathbf{P}_\rho$) and Spearman's ($\mathbf{S}_\rho$) coefficients, and their associated p-values (if greater than 0.005) for the correlations between the automatic metrics and human judgment under the G criterion.

correlations, with 0.89 and 0.85 for Pearson's and Spearman's ρ , respectively. At sentence level, the correlation between G_{score} and human judgment was moderate, with Pearson and Spearman ρ of 0.55 and 0.53, respectively. However, Evaluating G is undoubtedly a complex process influenced by various factors, making the evaluation challenging and increasing the likelihood of incorrect assessments.

For the M criterion, BLEU and BERTscore exhibited very strong correlations with human judgments, outperforming other automatic metrics. At the sentence level, BERTscore achieved the highest correlation with human judgment, with Pearson's and Spearman's correlation coefficients of 0.84 and 0.83, respectively. M_{score} came second, with Pearson's and Spearman's correlation coefficients of 0.79 and 0.75, respectively.

At the model level, BLEU demonstrated remarkable performance, with Pearson's and Spearman's correlation coefficients of 0.98 and 0.95, respectively. M_{score} also showed a very strong correlation with human judgment, scoring 0.94 for both Pearson and Spearman correlation coeffi-

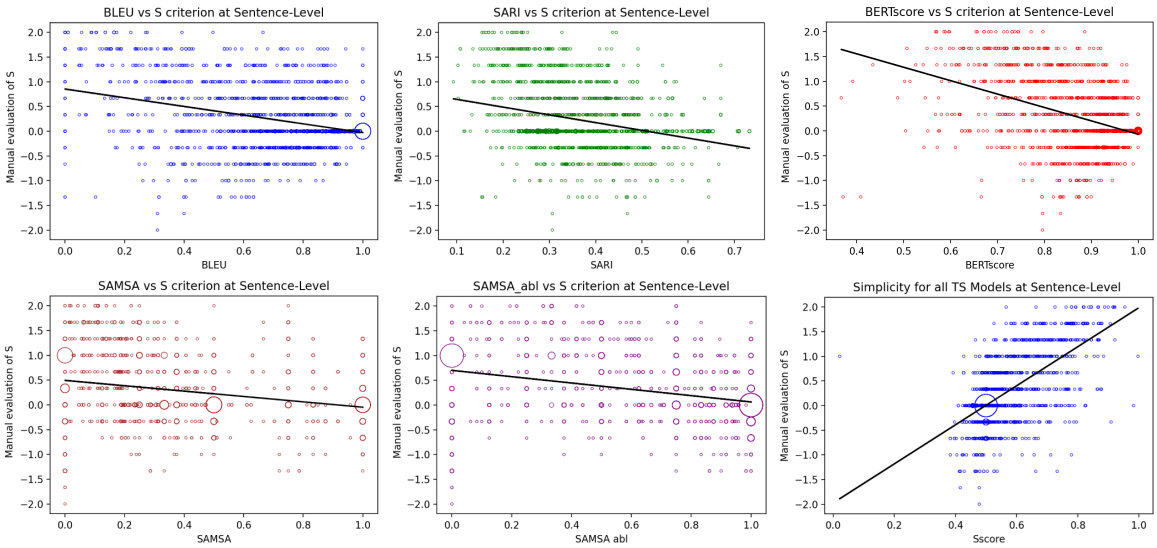

Figure 1: Scatter plots and regression lines at the sentence-level, depicting the relationship between the human scores assigned for S criterion (Y-axes) and the corresponding automatic model scores (X-axes). Each data point on the graph represents a sentence. (The size of each data point corresponds to the number of repetitions)

Metric	*Sentence-Level*		*Model-Level*	
	$\mathbf{P}_\rho$	$\mathbf{S}_\rho$	$\mathbf{P}_\rho$	$\mathbf{S}_\rho$
BLEU	0.77	0.75	**0.98**	**0.95**
SARI	0.36	0.24	0.79	0.63
BERTscore	**0.84**	**0.83**	0.97	0.94
SAMSA	0.45	0.49	0.72	0.77
SAMSA$_{abl}$	0.58	0.64	0.80	0.79
M$_{score}$	0.79	0.75	0.94	0.94

Table 9: The Pearson's ($\mathbf{P}_\rho$) and Spearman's ($\mathbf{S}_\rho$) coefficients, and their associated p-values (if greater than 0.005) for the correlations between the automatic metrics and human judgment under the M criterion.

cients. This is a notable achievement, considering that M_{score} is a reference-less metric, while both BLEU and BERTscore rely on the estimation based on 8 well-prepared references. In comparison to SAMSA and SAMSA$_{abl}$, the reference-less metrics in our experiment, M_{score} exhibited impressive performance (see Table 9).

In terms of overall quality, we conducted a comprehensive comparison of the automatic metrics against overall quality, which we calculated using two methods: first, by following AlAjlouni et al. methodology, utilizing *'Standard A'* (F$_A$); second, by employing the more conventional approach of relying on the arithmetic mean of M, G, and S scores (F$_{avg}$).

At the sentence-level, BLEU, BERTscore, SAMSA$_{abl}$, and CE_{score} exhibited a moderate correlation with overall quality, whether by F$_A$ or F$_{avg}$, as shown in Table 10. BERTscore took the lead, followed closely by BLEU, and CE_{score} followed in third place. Notably, the margins between their performance were quite small, and CE_{score} even outperformed the others in some cases (see Table 10).

At the model-level, all metrics (except SARI and SAMSA) showed very strong correlations with overall quality, whether calculated using F$_A$ or F$_{avg}$. Notably, CE_{score} displayed an impressive correlation with F$_A$, achieving coefficient values of 0.95 and 0.98 for Pearson and Spearman, respectively. The correlation with F$_{avg}$ was also substantial, with values of 0.94 and 0.95 for Pearson and Spearman, respectively. These results underscore the effectiveness of using CE_{score} as a robust metric for evaluating TS at the model-level. Figure 2 showcases scatter plots at the model-level between automatic metrics and F$_A$, where each point represents an individual model. The regression lines are included in these graphs for reference.

6 Conclusion

In conclusion, this research addresses the critical need for precise and efficient evaluation metrics in the field of text simplification (TS). Traditional evaluation metrics, originally designed for machine translation tasks, fall short in capturing the unique goals and challenges of TS. As a response to these limitations, we introduce CEScore, a Confidence Estimation Score model that directly evaluates the

Metric	*Sentence-Level*				*Model-Level*			
	Pearson$_\rho$		**Spearman**$_\rho$		**Pearson**$_\rho$		**Spearman**$_\rho$	
	$\mathbf{F}_A$	$\mathbf{F}_{avg}$	$\mathbf{F}_A$	$\mathbf{F}_{avg}$	$\mathbf{F}_A$	$\mathbf{F}_{avg}$	$\mathbf{F}_A$	$\mathbf{F}_{avg}$
BLEU	0.61	0.55	0.56	0.63	0.88	0.85	0.85	0.83
SARI	0.23	0.13	0.17	0.04	0.52	0.38 (0.05)	0.31 (0.12)	0.21 (0.29)
BERTscore	**0.62**	0.56	**0.56**	**0.64**	0.81	0.75	0.82	0.79
SAMSA	0.30	0.29	0.32	0.28	0.34 (0.09)	0.29 (0.15)	0.43 (0.03)	0.48 (0.01)
SAMSA$_{abl}$	0.43	0.42	0.46	0.37	0.72	0.73	0.78	0.81
CE$_{score}$	0.60	**0.57**	0.53	0.59	**0.95**	**0.94**	**0.98**	**0.95**

Table 10: The Pearson's ($\mathbf{P}_\rho$) and Spearman's ($\mathbf{S}_\rho$) coefficients, along with their associated p-values (if greater than 0.005), represent the correlations between the automatic metrics and the overall quality based on human judgments, examined at both the sentence and model levels. The overall quality was calculated using two methods: AlAjlouni's 'Standard A' ($\mathbf{F}_A$) or the arithmetic mean ($\mathbf{F}_{avg}$). The highest score in each column is denoted in bold.

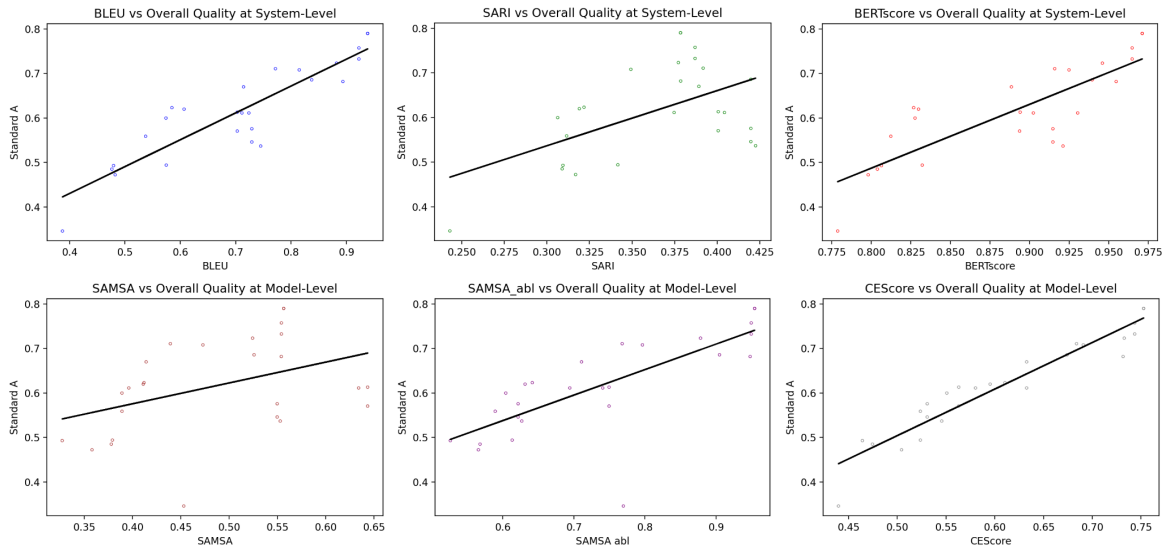

Figure 2: This figure includes scatter plots with regression lines as references at the model-level, illustrating the relationship between overall quality as calculated by FA (Y-axes) and the corresponding scores of the automatic models (X-axes). Each data point on the graph represents an individual model.

quality of simplified text in terms of simplicity (S), meaning preservation (M), and grammaticality (G). CEScore's comprehensive approach aligns with human evaluation and eliminates the reliance on reference texts, making it a valuable and contextually relevant tool for TS evaluation.

Our research contributes to the field in several key aspects. We have introduced statistical functions and innovative formulas for evaluating simplicity, including *Sentence Length Score* (SLS), *Average Sentence Familiarity* (ASF), and *Text Simplicity Score* (TSS). These formulas provide valuable insights into the simplicity of texts. Additionally, CEScore itself, which includes S_{score}, G_{score}, M_{score}, and CE_{score}, demonstrates strong correlations with human judgments. Particularly at the model level, it achieves remarkable coefficient values, making it a reliable and effective evaluation metric for TS models.

Moving forward, future work will focus on enhancing the accuracy of G_{score} and M_{score} by incorporating contextual embeddings. This step will address their sensitivity to lexical TS, further improving their ability to evaluate TS models accurately. These developments will contribute to the ongoing advancement of the field of TS and evaluation, ultimately making digital content more accessible to a broader audience.

References

Roee Aharoni and Yoav Goldberg. 2018. Split and rephrase: Better evaluation and stronger baselines.

In *the 56th Annual Meeting of the Association for Computational Linguistics*.

AlMotasem Bellah AlAjlouni and Jinlong Li. 2023. Knowledge transfer to solve split and rephrase. In *2023 International Conference on Information Technology*, pages 680–685. IEEE.

AlMotasem Bellah AlAjlouni, Jinlong Li., and Mo'ataz A. Ajlouni. 2023. Towards a comprehensive metric for evaluating text simplification systems. In *14th International Conference on Information and Communication Systems*.

Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. 2019. Easse: Easier automatic sentence simplification evaluation. In *the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 49–54.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2020. Data-driven sentence simplification: Survey and benchmark. *Computational Linguistics*, 46(1):135–187.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *the 20th international conference on computational linguistics*, pages 315–321.

Jan A. Botha, Manaal Faruqui, John Alex, Jason Baldridge, and Dipanjan Das. 2018. Learning to split and rephrase from wikipedia edit history. In *the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 732–737.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3):904–911.

Daniel Hershcovich, Omri Abend, and Ari Rappoport. 2017. A transition-based directed acyclic graph parser for ucca. In *In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1127–1138.

Van Heuven, Walter JB, Pawel Mandera, Emmanuel Keuleers, and Marc Brysbaert. 2014. Subtlex-uk: A new and improved word frequency database for british english. *Quarterly journal of experimental psychology*, 67(6):1176–1190.

Jacob D. M. C. Kenton and Lee K. Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

André FT Martins, Marcin Junczys-Dowmunt, Fabio N. Kepler, Ramón Astudillo, Chris Hokamp, and Roman Grundkiewicz. 2017. Pushing the limits of translation quality estimation. *Transactions of the Association for Computational Linguistics*, 5:205–218.

Shashi Narayan and Claire Gardent. 2014. Hybrid simplification using deep semantics and machine translation. In *In 52th Annual Meeting of the Association for Computational Linguistics*, page 435–445.

Shashi Narayan, Claire Gardent, Shay B. Cohen, and Anastasia Shimorina. 2017. Split and rephrase. In *Conference on Empirical Methods in Natural Language Processing*, pages 617–627.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2019. Transforming complex sentences into a semantic hierarchy. In *the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3415–3427.

Sergiu Nisioi, Sanja ˇ Stajner, Simone Paolo Ponzetto, and Liviu P. Dinu. 2017. Exploring neural text simplification models. In *In 55th Annual Meeting of the Association for Computational Linguistics*, page 85–91.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *the 40th annual meeting of the Association for Computational Linguistics*, page 311–318.

Advaith Siddharthan. 2002. An architecture for a text simplification system. In *Language Engineering Conference*, pages 64–71. IEEE.

Lucia Specia, Fr´ed´eric Blain, Varvara Logacheva, Ram´on Astudillo, and Andr´e F. T. Martins. 2018. Findings of the wmt 2018 shared task on quality estimation. In *the Third Conference on Machine Translation*, page 689–709.

Lucia Specia, Marco Turchi, Nicola Cancedda, Nello Cristianini, and Marc Dymetman. 2009. Estimating the sentence-level quality of machine translation systems. In *the 13th Annual conference of the European Association for Machine Translation*.

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

Elior Sulem, Omri Abend, and Ari Rappoport. 2018b. Semantic structural evaluation for text simplification. In *NAACL-HLT*, pages 685–696.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018c. Simple and effective text simplification using semantic and neural methods. In *the 56th Annual Meeting of the Association for Computational Linguistics*, pages 162–173.

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.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.