Context and Literacy Aware Learnable Metric for Text Simplification

Jeongwon Kwak[1,2], **Hyeryun Park**[1,2], **Kyungmo Kim**[1], **Jinwook Choi**[2,3,4]

[1]Interdisciplinary Program for Bioengineering, Graduate School, Seoul National University
[2]Integrated Major in Innovative Medical Science, Graduate School, Seoul National University
[3]Department of Biomedical Engineering, College of Medicine, Seoul National University
[4]Medical Research Center, Institute of Medical and Biological Engineering,
Seoul National University
{jeongwonkwak17,hyerpark1115}@gmail.com, {medinfoman,jinchoi}@snu.ac.kr

Abstract

Automatic evaluation of text simplification is important; but assessing its transformation into simpler sentences can be challenging for various reasons. However, the most commonly used metric in text simplification, SARI (Xu et al., 2016), fails to capture the difficulty of generating words that are not present in the references, regardless of their meaning. We propose a new learnable evaluation metric that decomposes and reconstructs sentences to simultaneously measure the similarity and difficulty of sentences within a single system. Through experiments, we confirm that it exhibits the highest similarity in correlation with the human evaluation.

1 Introduction

Text simplification refers to the task of transforming sentences into easily understandable sentences while maintaining context (Saggion and Hirst, 2017). This is effective in various domains such as education and biomedicine. In the biomedical field, in particular, there is a need to simplify health information which is often written in a complex manner that is difficult for the general public to understand (Brach and Harris, 2021; van den Bercken et al., 2019). However, it is challenging to evaluate whether complex sentences have been simplified successfully. This is because the process of converting a complex sentence into a simpler one involves various operations such as information deletion, paraphrasing, and insertion, while ensuring that the semantic meaning remains equivalent. Thus, most existing metrics (Kincaid et al., 1975; Papineni et al., 2002; Xu et al., 2016; Zhang* et al., 2020) evaluate text simplification by separately assessing how semantically similar the output is and how effectively it has been simplified. Among metrics commonly used in current systems, SARI is the most popular. SARI measures the degree of n-gram overlap and evaluates aspects such as information preservation, deletion, and insertion. However, it fails to capture word-level differences when words that are not present in a reference are generated (see Figure 1). Additionally, another metric, BERTScore (Zhang* et al., 2020), also falls short in evaluating sentence-level quality because

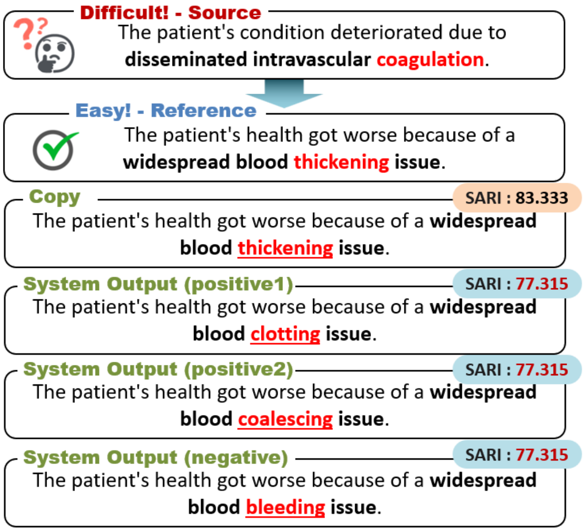
Difficult! - Source
The patient's condition deteriorated due to **disseminated intravascular coagulation**.
Easy! - Reference
The patient's health got worse because of a **widespread blood thickening issue**.
Copy
SARI : 83.333
The patient's health got worse because of a **widespread blood thickening issue**.
System Output (positive1)
SARI : 77.315
The patient's health got worse because of a **widespread blood clotting issue**.
System Output (positive2)
SARI : 77.315
The patient's health got worse because of a **widespread blood coalescing issue**.
System Output (negative)
SARI : 77.315
The patient's health got worse because of a **widespread blood bleeding issue**.

Figure 1: The score variations of metrics for sentences generated by systems such as GPT-3.5 (Ouyang et al., 2022), in comparison to a copy version of the reference. "Widespread blood thickening issue" marked in bold is a term simplified for "disseminated intravascular coagulation," which means a severe condition where blood clots form excessively in small blood vessels. Changing "thickening" to "clotting" or "coalescing", both of which are not in the reference, do not affect the SARI score. Also, "bleeding" which has the opposite meaning of "clotting" results in the same outcome.

Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM), pages 175–180
December 6, 2023 ©2023 Association for Computational Linguistics

it considers both complex and simplified sentences to be semantically similar.

To address this issue, we introduce learnable evaluation metrics. Our metric not only evaluates whether the generated text is semantically flawless but also comprehensively assesses the literacy level of the text. To train our model, we leverage the first supervised metric, LENS (Maddela et al., 2023), introduced along with the training dataset SIMPEVAL$_{\text{PAST}}$. In this metric, we train modules to evaluate the literacy level of the generated text, assessed its semantic similarity with the original sentence, and compare it with references. The experimental findings show an improvement in performance over conventional systems.

2 Background

SARI measures the overlap between source sentences, system output sentences, and reference sentences based on n-grams. It considers three aspects: how much information has been deleted (del), how much new information has been added (add), and how well existing information has been retained (keep). However, a caveat may exist in this regard. If the output sentence generates words that are not present in the reference, they may not be detected even though these words could potentially hinder the quality of the sentence.

The LENS collected SIMPEVAL$_{\text{PAST}}$, which includes 12K human ratings of 2.4K simplifications generated by 24 different systems applied to sentences from TurkCorpus (Xu et al., 2016) for training LENS. They selected 100 challenging sentences which were used in the training process of the model from TurkCorpus and ASSET (Alva-Manchego et al., 2020). ASSET provides 10 human references for each complex sentence. To evaluate the performance of the model and other simplification metrics, SIMPEVAL$_{2022}$ comprising 1K human ratings of 360 simplifications generated by human annotators and state-of-the-art models was introduced. WIKI-DA (Alva-Manchego et al., 2021) composes evaluations of 600 sentences generated by 6 different systems, assessing fluency, preservation, and simplicity.

The effectiveness of learnable evaluation metrics has been demonstrated in machine translation (Sellam et al., 2020; Rei et al., 2020). LENS, for the first time, applies learnable evaluation to text simplification. Their model encodes all text components (input texts, system outputs, and references) into RoBERTa (Liu et al., 2019) embeddings, combines them, and feeds them into a feedforward network to predict the scores. These scores are then compared with human rating scores and the mean squared error (MSE) loss is calculated.

3 Methods

In our model, we introduce context-aware and literacy-aware layers to comprehensively assess the similarity and complexity of sentences. As shown in Figure 2, the context module evaluates whether the system output sentence is generated in a manner similar to the meaning of the input and reference sentences. The literacy module assesses the literacy level of the system output sentences and learns whether the literacy level of the system's output sentence is easier than that of the input sentence, or similar to the literacy level of the reference sentence. To achieve this, the context and literacy-aware modules have distinct loss functions. Furthermore, we concatenate the vectors extracted from these modules and calculate the final loss function based on their differences from the original vector.

Given a source text s, the corresponding system output o and references $R = \{r_1, r_2, ..., r_n\}$, our model predicts the quality scores of system outputs for each reference and selects the top k ($k \leq n$) scores in descending order. (s, o, r_i) are encoded by a Transformer-based encoder represented as embeddings (e_s, e_o, e_r). In the following section, two modules that fed into these embeddings are described.

3.1 Context-Aware Module

The context-aware layer which is expected to perceive context in each embedding feeds (e_s, e_o, e_r) to generate (e_s^C, e_o^C, e_r^C).

$$e_s^C = W_s^c e_s + b_s^c, e_o^C = W_o^c e_o + b_o^c, e_r^C = W_r^c e_r + b_r^c,$$

where $W_s^c, W_o^c, W_r^c \in R^{H \times H'}$ and $b_s^c, b_o^c, b_r^c \in R^{H'}$. H is the embedding dimension of the encoder, H' is the hidden dimension, α represents the cosine similarity loss, and $\alpha(u, v, n)$ generates a loss based on the difference between the cosine similarity of u and v, and the value of n ($-1 \leq n \leq 1$).

$$L_{cnt} = \alpha(e_s^C, e_o^C, 1) + \alpha(e_s^C, e_r^C, 1) + \alpha(e_o^C, e_r^C, 1)$$

In this module, the embeddings of the three elements are trained to be positioned closely in the

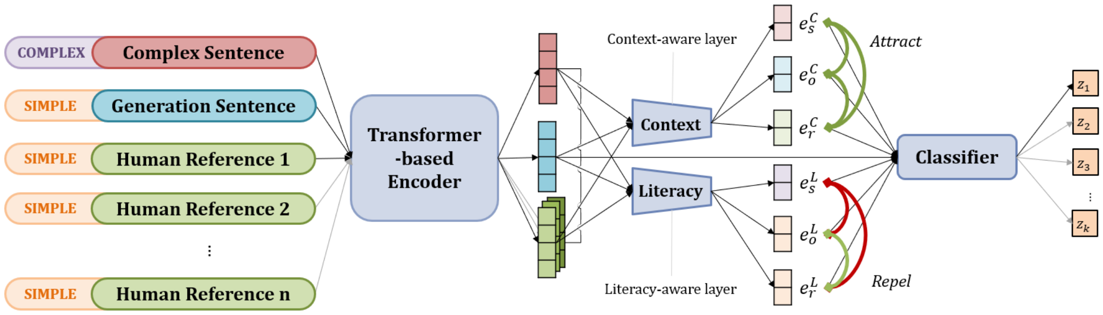

Figure 2: The overall architecture of the model, which includes context-aware and literacy-aware layers to measure these two aspects for text simplification. Embeddings passed through both modules are subjected to different loss function techniques. The red lines in the figure signify that the interconnected vectors are trained to point in opposite directions, while the green lines indicate that they are trained to point in the same direction. We aim to separate the aspects in which the output sentence aligns with the other two sentences (*contextual similarity*) and the aspects in which the source sentence differs from the other two sentences (*literacy-level*). These extracted vectors also reflect the difference from the original embeddings passed through a transformer-based encoder as part of the loss.

vector space because their contexts are expected to be similar. We compute the sum of the cosine similarities for each embedding denoted by L_{cnt}.

3.2 Literacy-Aware Module

This layer operates in coordination with the context-aware module by feeding it the same vector and producing (e_s^L, e_o^L, e_r^L) in a manner identical to that of the module.

$$e_s^L = W_s^l e_s + b_s^l, e_o^L = W_o^l e_o + b_o^l, e_r^L = W_r^l e_r + b_r^l,$$

where $W_s^l, W_o^l, W_r^l \in R^{H \times H'}$ and $b_s^l, b_o^l, b_r^l \in R^{H'}$.

Excluding the embeddings of the source sentences, all the other sentence embeddings likely to have a simple level of difficulty. Thus, we hope that e_s^L and e_o^L indicate in different directions, similarly for e_s^L and e_r^L , for which we assign n the value -1 as target cosine similarity score. However, for e_o^L and e_r^L, we wish them to have the same literacy level embeddings; therefore, we assign a value of 1.

$$L_{lit} = \alpha(e_s^L, e_o^L, -1) + \alpha(e_s^L, e_r^L, -1) + \alpha(e_o^L, e_r^L, 1)$$

In both modules, we do not include the cosine similarity between their own embeddings in the loss function, because it is equal to 1.

3.3 Reconstruction Module

We introduce a new module that restores the embeddings passed through the context-aware module and the embeddings passed through the literacy-aware module to match the original embeddings. The symbol [;] represents concatenation and it is used to concatenate the embeddings that have passed through the preceding two modules.

$$e_s^{'} = [e_s^C; e_s^L],\ e_o^{'} = [e_o^C; e_o^L],\ e_r^{'} = [e_r^C; e_r^L]$$

Afterwards, we reshape them to match the dimensions of the original vectors with ($e_s^{'}$,$e_o^{'}$,$e_r^{'}$) being transformed into ($e_s^{''}$,$e_o^{''}$,$e_r^{''}$), respectively.

$$e_s^{''} = W_s e_s^{'} + b_s,\ e_o^{''} = W_o e_o^{'} + b_o,\ e_r^{''} = W_r e_r^{'} + b_r,$$

where $W_s, W_o, W_r \in R^{H' \times H}$ and $b_s, b_o, b_r \in R^H$.

We define L_{rct} by adding the difference with the original vectors individually, and use the Mean Squared Error (MSE) function for β.

$$L_{rct} = \beta(e_s, e_s^{''}) + \beta(e_o, e_o^{''}) + \beta(e_r, e_r^{''})$$

3.4 Adaptable Score Based on Ratings

We implement a novel train method to establish dependency on the rating scores of generated sentences. We assign target cosine similarity scores (CS_t) at L_{lit} differently based on how well the generated sentences align with text simplification. The similarity scores between s and o, as well as between s and r, are equal, while the scores between o and r are opposite of their scores. s and r have opposing levels of literacy aspect. We aim to assign CS_t by separating the ratings of training data into three parts, assuming it falls within the range of $-\sigma$ to σ as z-scores. We assign $\sigma = 2$.

First, if the real rating score x is greater than or equal to σ, it refers that the model has generated sentences effectively for an easy level

	SIMPEVAL$_{2022}$				**WIKI-DA**		
	$\tau_{del}\uparrow$	$\tau_{para}\uparrow$	$\tau_{spl}\uparrow$	$\tau_{all}\uparrow$	**Fluency↑**	**Meaning↑**	**Simplicity↑**
FKGL	-0.25	-0.556	-0.31	-0.356	0.054	0.145	0.001
BLEU	0.0	0.048	-0.054	-0.033	0.46	0.622	0.438
SARI	0.0	0.206	0.14	0.149	0.335	0.534	0.366
BERTScore	-0.25	0.238	0.093	0.112	0.636	0.682	0.614
LENS	-0.5	0.429	0.333	0.331	0.807	0.668	0.749
LENS†	-0.5	**0.27**	0.24	0.228	**0.781**	**0.681**	**0.723**
$Ours_{E+R}$	0.0	0.206	0.31	0.283*	0.674	0.48	0.537
$Ours_{E+R+S}$	0.0	0.246	**0.359***	**0.328***	0.693	0.518	0.581

Table 1: We evaluate the SIMPEVAL$_{2022}$ dataset and WIKI-DA using both conventional metrics and the existing learnable evaluation metric. The dataset is annotated with deletions, paraphrases, and splittings based on how system output is generated, and we apply Kendall's Tau (τ) coefficient for the three aspects as well as an overall assessment. We present the Pearson correlation coefficients for WIKI-DA across three dimensions. The † indicates the reimplementation of LENS, and although we follow the parameter settings disclosed in LENS, there is a discrepancy of 0.103. The subscript E in our model denotes the results after passing through the two Extract modules, while $E + R$ represents the outcomes when Reconstruction is also performed. The subscript S denotes the results using an adaptable score based on ratings. Except LENS, the best is marked in bold. The $*$ indicates a statistically significant difference with that p-values less than 0.05.

of literacy. As s and o are considered to have opposite difficulty levels, the cosine similarity between e_s^L and e_o^L is trained to be -1. Second, if x is lower than or equal to $-\sigma$, it refers that the model do not perform effectively for an easy level of it. Also, s and o have a similar level of difficulty, CS_t is assigned to 1. Finally, for ratings falling within the range between σ and $-\sigma$, CS_t is assigned to $-\frac{x}{\sigma}$ by mapping the range of ratings for the generated sentences to the range of the target cosine similarity scores.

$$CS_t = \begin{cases} -1 & \text{if } x \geq \sigma \\ 1 & \text{if } x \leq -\sigma \\ -\frac{x}{\sigma} & \text{otherwise} \end{cases}$$

Given that the rating score x is greater than the σ, we could regard the generated sentences as having a similar and relatively easy level compared to the reference. For example, if x = 3 and σ = 2, the cosine similarity scores between e_s^L and e_o^L should be -1, while scores between e_o^L and e_r^L should be 1.

3.5 Integration of Embeddings

We incorporate embedding E_v which passed through the newly introduced modules based on the embedding E_u used in the existing LENS.

$$E_u = [s; o; r_i; o \odot s; o \odot r_i; |o - s|; |o - r_i|]$$

$$E_v = [e_s^C; e_o^C; e_r^C; e_s^L; e_o^L; e_r^L],\ E = [E_u; E_v]$$

The embedding E is subsequently input into a feedforward network for the prediction of z_i. The MSE loss is calculated as L_{fcn} between z_i and the corresponding human ranking score. Finally, we calculate the loss denoted as L_{tot}.

$$L_{tot} = L_{cnt} + L_{lit} + L_{rct} + L_{fcn}$$

4 Experiments

4.1 Experimental Setup

We evaluate the proposed method using a text simplification benchmark. We train the model using the SIMPEVAL$_{\text{PAST}}$ dataset and evaluate it using on the SIMPEVAL$_{2022}$ dataset, as detailed in Section 2. The dataset is categorized into three types for each system: deletions, paraphrases, and splittings. To assess the model's performance, we employ Kendall's Tau coefficient τ ($-1 \leq \tau \leq 1$) as the evaluation metric. For each complex sentence, the trends in the human-rated and model-predicted scores of the two systems are compared. If the trends match, they are considered concordant; otherwise, they are considered discordant. The tau coefficient is calculated by dividing the difference between the number of concordant and discordant pairs by the sum of the concordant and discordant pairs. For consistency, we utilize the training parameters provided with the dataset. For more details on the experimental setup, see Appendix A.

4.2 Results

As shown in Table 1, the conventional non-learnable metrics tend to have either negative or small overall Kendall's Tau values. Learnable metrics, on the other hand, generally demonstrate slightly improved performance, with especially higher values in the splitting aspect compared to the reimplementation of LENS. For the examples of the results using $Ours_{E+R+S}$, see Appendix B. In the WIKI-DA, $Ours$ shows a decrease compared to the reimplementation of LENS, but we confirm that $Ours_{E+R+S}$ exhibits an improvement over $Ours_{E+R}$. For the SIMPEVAL$_{2022}$, $Ours_{E+R}$ demonstrates an improvement of 0.055 compared to the reimplemented LENS, while $Ours_{E+R+S}$ shows a 0.109 enhancement over $Ours_{E+R}$.

5 Conclusion

We have proposed a new decomposition-guided learnable evaluation metric for text simplification, an automatic metric capable of simultaneously evaluating semantic preservation and literacy levels in text simplification tasks. We succeed in measuring these two aspects separately, as they demonstrates a higher correlation with human evaluations than existing automatic metrics. This approach can be extended to transfer tasks with binary classes and holds promise for application to datasets with diverse literacy levels.

Limitations

The proposed system is primarily focused on the sentence-level; but there is a need to extend it to handle document-level data such as medical documents. In addition, we have only addressed binary levels of simplicity, it is necessary to expand the model to accommodate datasets that cover various levels of granularity. Also, our research is limited to English; therefore, research in other languages should be conducted.

Acknowledgements

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education [No. NRF-2021R1I1A4A01042182]

References

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4668–4679, Online. Association for Computational Linguistics.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2021. The (un) suitability of automatic evaluation metrics for text simplification. *Computational Linguistics*, 47(4):861–889.

Cindy Brach and Linda M Harris. 2021. Healthy people 2030 health literacy definition tells organizations: make information and services easy to find, understand, and use. *Journal of general internal medicine*, 36(4):1084–1085.

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. *Research branch report*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408, Toronto, Canada. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

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.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Horacio Saggion and Graeme Hirst. 2017. *Automatic text simplification*, volume 32. Springer.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Laurens van den Bercken, Robert-Jan Sips, and Christoph Lofi. 2019. Evaluating neural text simplification in the medical domain. In *The World Wide Web Conference*, WWW '19, page 3286–3292, New York, NY, USA. Association for Computing Machinery.

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. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*.

A Experimental Details

We employed 20% of the SimpEval$_{\text{PAST}}$ dataset for validation, with a value of k set to 3. Our batch size was set at 2, and the hidden size of both the two modules and the feedforward network was 384. We applied a dropout rate of 0.5 and utilized a learning rate of 3.1e-05, while the encoder learning rate was set to 1.0e-05. In addition, we selected the checkpoint with the lowest loss when training up to 10 epochs. We train the model on two NVIDIA GeForce RTX 3090 GPU, which takes around 1 hour 30 minutes.

B Additional Examples

Example1(Splitting):

Complex: In addition to farming, corbin has expanded his business to include agritourism, using his farm to host weddings, host corporate meetings, and attract visitors to a large annual christmas-themed display.

Reference: Corbin has expanded his farming business to include agritourism. He uses his farm to host weddings, corporate meetings, and a large annual christmas-themed display.

System Output: Corbin's business has grown beyond farming to include agritourism. This involves hosting weddings, corporate meetings, and a big holiday celebration open to visitors.

LENS†: 59.51 **Ours:** 78.76 **Human:** 80.33

Example2(Splitting):

Complex: The film stars julie ledru as julia, a young woman living in social housing who has a poor relationship with her mother; passionate about the sport of motocross, she tries to gain the opportunity to participate in the sport by posing as a buyer on shopping websites so that she can take bikes out for test rides.

Reference: In this film, julie ledru acts as a young woman named julia. Julia lives in social housing and does not have a good relationship with her mother. She loves the sport of motocross and wants to find a way to take part in it. Hence, she acts as a buyer on shopping websites so that she can take bikes out for test rides.

System Output: Julie ledru portrays julia in the film; julia lives in public housing and her relationship with her mother is strained. She loves motocross and will do whatever it takes to try it out. to this end, she pretends to be a buyer on shopping websites in order to take bikes out for test rides.

LENS†: 54.74 **Ours:** 63.44 **Human:** 63.33

Example3(Paraphrase):

Complex: A morning consult poll in mid-november 2022 found that only 28% of americans supported having an impeachment investigation or investigations into hunter biden's business dealings to be a top congressional priority, though it also found more than half of self-identified republicans supported it as a top congressional priority.

Reference: A poll found that only 28% of americans supported having an impeachment investigation into hunter biden, but it also found half of republicans supported it.

System Output: In mid-november 2022, a morning consult poll revealed that only 28% of americans thought an impeachment investigation or investigations into hunter biden's business dealings should be a major congressional focus, but over half of republicans agreed it should be a priority.

LENS†: 38.38 **Ours:** 43.08 **Human:** 84.33

Example4(Paraphrase):

Complex: On november 9, fetterman was projected as the winner of the election by media outlets.

Reference: On novermber 9, media outlets predicted fetterman would win the us senate election in pennsylvania.

System Output: On november 9, the media said that fetterman was the winner of the election.

LENS†: 57.23 **Ours:** 40.49 **Human:** 66.67