FRUSTRATINGLY SIMPLE PROMPTING-BASED TEXT DENOISING

Jungyeul Park*
Department of Linguistics
The University of British Columbia
Vancouver, BC V6T 1Z4, Canada
jungyeul@mail.ubc.ca

Mengyang Qiu
Department of Psychology
Trent University
Peterborough, ON K9L 0G2, Canada
mengyangqiu@trentu.ca

ABSTRACT

This paper introduces a novel perspective on the automated essay scoring (AES) task, challenging the conventional view of the ASAP dataset as a static entity. Employing simple text denoising techniques using prompting, we explore the dynamic potential within the dataset. While acknowledging the previous emphasis on building regression systems, our paper underscores how making minor changes to a dataset through text denoising can enhance the final results.

1 INTRODUCTION

Text denoising is a crucial step in natural language processing (NLP) and text analysis tasks (Sun & Jiang, 2019; Xian et al., 2021). One of its major applications is in optical character recognition (OCR). Recently, the technique has also found utility in image caption editing, where computer vision and natural language processing intersect (Wang et al., 2022; Yuan et al., 2023). Despite its significance, in written text processing, the task of text denoising has often been overlooked due to the perception that the provided datasets are already well-prepared. Prompting has proven successful in the text transfer task, even with small language models (Suzgun et al., 2022). Therefore, we approach the processing of written text as a form of text transfer, employing prompting.

In the text denoising experiment, our objective is to *clean* the dataset used for automated essay scoring (AES) in preparation for the linear regression task. AES involves the use of an automated system to assign numerical grades or scores to essays written in a first language (L1) within an educational setting. It is crucial to note that English AES systems rely on the ASAP dataset, initially provided in the Kaggle competition of 2012.[1] This dataset has become widely adopted in various AES systems (Alikaniotis et al., 2016; Cummins et al., 2016; Dong et al., 2017; Ridley et al., 2021; Chen & Li, 2023; Do et al., 2023). The original ASAP dataset employed a named entity recognition approach using Stanford CoreNLP (Manning et al., 2014) and a range of pattern matching rules to eliminate personally identifying information from the essays. Consequently, entities within the text are identified and replaced with strings initiated with the '@' symbol, such as `@PERSON1`. Furthermore, more than 5% of sentences exhibit encoding issues where UTF-8 symbols are not correctly displayed, in addition to the presence of non-word entities.

We classify sentences containing encoding issues and non-word entities as noise, and subsequently, undergo a denoising process to address them. Our hypothesis posits that enhancing text quality through denoising will yield improved linear regression results in AES. Throughout this paper, we use the term **prompt** in two distinct contexts: (1) In the context of essay writing, a *prompt* refers to a specific question, topic, or statement serving as the starting point for the essay. (2) In the context of generative pre-trained transformers, a *prompt* denotes the input or stimulus provided to the model to generate a response.

*This work was supported in part by Oracle Cloud credits and related resources provided by Oracle for Research.

[1] https://www.kaggle.com/c/asap-aes

2 TEXT DENOISING

In the context of the ASAP dataset, Figure 1 illustrates an instance of text denoising. For this process, we employ two prompts with `gpt-3.5-turbo-instruct`: one to address encoding errors and another to replace non-word entities with arbitrary entity names sequentially.[2] It is noteworthy that `gpt-3.5-turbo-instruct` has a tendency to correct grammatical errors in the original text during sentence generation. To restore the original words, we utilize the `.m2` annotation generated by ERRANT (Bryant et al., 2017). This annotation delineates the modifications made in the original text, allowing us to retain only the symbols with corrected encoding and the replaced non-word entities.

ORIGINAL TEXT:	`people get @CAPS2 addicted that they don<U+0092>t exercize and become obeast, ...`
CLEANED TEXT:	`people get too addicted that they don't exercize and become obeast, ...`

Figure 1: Example of text denoising: the Unicode symbol `U+0092` is replaced with `'`, and `@CAPS2` is substituted with an arbitrary word

3 EXPERIMENTS AND RESULTS

After text denoising, we utilize `roberta-base` (Liu et al., 2019) for the linear regression task to derive the overall score. The results of the linear regression by quadratic weighted kappa (QWK), along with perplexity values, are presented in Table 1 for both the original and cleaned texts. It's important to note that all linear regression evaluations are conducted on the original text, even when the model is trained using the cleaned text, ensuring a fair comparison. We employ prompt-based 8-fold cross-validation, considering ASAP's eight different prompt essay sets, and the results represent the average OVERALL scores.

	Sentence ppl	Token ppl	Regression
ORIGINAL TEXT	532.571	65.535	0.6047
CLEANED TEXT	520.023	51.097	0.6143

Table 1: Experiment results, including perplexity and regression using QWK

4 DISCUSSION AND CONCLUSION

The ASAP dataset is often regarded as a given, with prior research primarily concentrating on constructing a regression system. However, making even subtle modifications through denoising to the dataset can yield improved results. In the previous automatic writing evaluation system for L2 writing, Lim et al. (2023) observed that attention did not focus on spelling errors. Rather, a pre-trained large language model can be associated with more proper words instead of spelling errors to predict results. Through the cleaning process of the training dataset, we confirm their finding that the refined training dataset exhibits enhanced learning capabilities, leading to more accurate result prediction.

A closer examination of the prompt-by-prompt QWK results reveals insightful nuances. In most cases, the cleaned text surpasses the original text, underscoring the efficacy of the denoising process. While the effectiveness of text denoising is evident, the nuanced variations across prompts necessitate careful consideration in experimental design. As we explore these avenues, we anticipate that refining the dataset and incorporating additional features will contribute to the ongoing improvement of automated essay scoring systems. The limitation of our study still lies in the choice of employing a simple transformer for the regression task instead of building our own neural regression system. As a result, despite our inability to achieve state-of-the-art results for the ASAP dataset, as demonstrated by previous neural AES systems incorporating numerous linguistic features (Ridley et al., 2021; Chen & Li, 2023; Do et al., 2023), our approach yields relatively good results with minimal effort.

[2]The prompting process costs $6.87 for the entire 12,978 ASAP data entries, where we select sentences containing only such encoding errors and non-word entities.

URM STATEMENT

The authors acknowledge that both authors of this work meet the URM criteria of ICLR 2024 Tiny Papers Track.

REFERENCES

Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. Automatic Text Scoring Using Neural Networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 715–725, Berlin, Germany, 8 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1068. URL https://aclanthology.org/P16-1068.

Christopher Bryant, Mariano Felice, and Ted Briscoe. Automatic Annotation and Evaluation of Error Types for Grammatical Error Correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 793–805, Vancouver, Canada, 2017. Association for Computational Linguistics. URL http://aclweb.org/anthology/P17-1074.

Yuan Chen and Xia Li. PMAES: Prompt-mapping Contrastive Learning for Cross-prompt Automated Essay Scoring. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1489–1503, Toronto, Canada, 7 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.acl-long.83.

Ronan Cummins, Meng Zhang, and Ted Briscoe. Constrained Multi-Task Learning for Automated Essay Scoring. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 789–799, Berlin, Germany, 8 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1075. URL https://aclanthology.org/P16-1075.

Heejin Do, Yunsu Kim, and Gary Geunbae Lee. Prompt- and Trait Relation-aware Cross-prompt Essay Trait Scoring. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1538–1551, Toronto, Canada, 7 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.findings-acl.98.

Fei Dong, Yue Zhang, and Jie Yang. Attention-based Recurrent Convolutional Neural Network for Automatic Essay Scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pp. 153–162, Vancouver, Canada, 8 2017. Association for Computational Linguistics. doi: 10.18653/v1/K17-1017. URL https://aclanthology.org/K17-1017.

KyungTae Lim, Jayoung Song, and Jungyeul Park. Neural automated writing evaluation for Korean L2 writing. *Natural Language Engineering*, 29(5):1341–1363, 2023. doi: 10.1017/S1351324922000298. URL https://www.cambridge.org/core/journals/natural-language-engineering/article/neural-automated-writing-evaluation-for-korean-l2-writing/C1C7090E7DB7F0756CCD0B4B002E096A.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.1, 2019. URL http://arxiv.org/abs/1907.11692.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 55–60, Baltimore, Maryland, 6 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P14-5010.

Robert Ridley, Liang He, Xin-yu Dai, Shujian Huang, and Jiajun Chen. Automated Cross-prompt Scoring of Essay Traits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15): 13745–13753, 5 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17620.

Yifu Sun and Haoming Jiang. Contextual Text Denoising with Masked Language Model. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pp. 286–290, Hong Kong, China, 11 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-5537. URL https://aclanthology.org/D19-5537.

Mirac Suzgun, Luke Melas-Kyriazi, and Dan Jurafsky. Prompt-and-Rerank: A Method for Zero-Shot and Few-Shot Arbitrary Textual Style Transfer with Small Language Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 2195–2222, Abu Dhabi, United Arab Emirates, 12 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.141.

Zhen Wang, Long Chen, Wenbo Ma, Guangxing Han, Yulei Niu, Jian Shao, and Jun Xiao. Explicit Image Caption Editing. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision – ECCV 2022*, pp. 113–129, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-20059-5.

Jiayi Xian, Dingcheng Li, Alexander Hanbo Li, Derek Liu, Xing Fan, Edward Guo, Yang Liu, and Yuqing Tang. Unified denoising pretraining and finetuning for data and text. In *WeCNLP 2021*, Virtually, 2021. wecnlp.ai. URL https://www.amazon.science/publications/unified-denoising-pretraining-and-finetuning-for-data-and-text.

Mengqi Yuan, Bing-Kun Bao, Zhiyi Tan, and Changsheng Xu. Adaptive Text Denoising Network for Image Caption Editing. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 19(1s), 2 2023. ISSN 1551-6857. doi: 10.1145/3532627. URL https://doi.org/10.1145/3532627.

A APPENDIX

A.1 RESTORING THE ORIGINAL TEXT FROM GPT-CORRECTED TEXT

We use the following two prompts with `gpt-3.5-turbo-instruct` for text denoising:

1. `Copy the sentence by replacing utf8 encoding error characters into the correct ascii symbols`
2. `Copy the sentence by replacing @words into the real words`

After obtaining the text generated by `gpt`, the `errant_parallel` tool produces the `.m2` file. The annotations in the `.m2` file reveal the results of the prompting process, which involves fixing encoding errors and replacing non-word entities.

```
S 682|1|2 I think this because mere cases of suicide are from online bullying,
people get @CAPS2 addicted that they don<U+0092>t exercize and become obeast,
and because it is bad for the environment.
A 0 1|||U:OTHER||||||REQUIRED|||-NONE-|||0
A 19 20|||R:VERB|||don't|||REQUIRED|||-NONE-|||0        (***)
A 20 21|||R:SPELL|||exercise|||REQUIRED|||-NONE-|||0
A 23 24|||R:OTHER|||obese,|||REQUIRED|||-NONE-|||0
```

```
S 682|1|2 I think this because mere cases of suicide are from online bullying,
people get @CAPS2 addicted that they don't exercize and become obeast, and
because it is bad for the environment.
A 0 1|||U:OTHER||||||REQUIRED|||-NONE-|||0
A 5 6|||R:ADJ|||many|||REQUIRED|||-NONE-|||0
A 10 11|||R:OTHER|||caused by|||REQUIRED|||-NONE-|||0
A 13 14|||R:NOUN|||individuals|||REQUIRED|||-NONE-|||0
A 15 16|||R:OTHER|||too|||REQUIRED|||-NONE-|||0         (***)
A 20 21|||R:SPELL|||exercise|||REQUIRED|||-NONE-|||0
A 23 24|||R:OTHER|||obese,|||REQUIRED|||-NONE-|||0
A 28 29|||R:ADJ|||harmful|||REQUIRED|||-NONE-|||0
A 29 30|||R:PREP|||to|||REQUIRED|||-NONE-|||0
```

where `S 682|1|2` represents `source`, the ASAP id, the prompt number, and the sentence id, respectively. We retain only the encoding errors and replace non-word entities in `As` (the annotations corrected by `gpt`) marked with `(***)`. ERRANT enables us to regenerate the corrected sentence using the corresponding annotation information: *people get too addicted that they don't exercize and become obeast, ...* as in the original text, instead of *people get too addicted that they don't exercise and become obese, ...* as `gpt` generated.

A.2 DETAILED EXPERIMENT RESULTS

Detailed prompt-by-prompt results are presented in Table 2, including the outcomes of encoding fixed without replacing non-word entities (ENCODING FIXED), the evaluation conducted on the corresponding dataset rather than using the original text (CLEANED'), and the evaluation conducted on the original text to ensure a fair comparison (CLEANED) as in Table 1. During training, we utilize the default values of the `Trainer` class,[3] employing the RoBERTa base model. We normalize the score to the range of 0 and 1, and we multiply the results by 100 to calculate QWK using the standard evaluation script provided by `ets.org`.

	Prompt 1	Prompt 2	Prompt 3	Prompt 4	Prompt 5	Prompt 6	Prompt 7	Prompt 8	AVERAGE
ORIGINAL	0.6187	0.6308	0.6962	0.6626	0.7158	0.5924	0.4622	0.4586	0.6047
ENCODING FIXED	0.6517	0.5855	0.6872	0.6290	0.7485	0.6174	0.4692	0.4943	0.6103
CLEANED'	0.7193	0.5414	0.6906	0.6454	0.7374	0.6187	0.4441	0.4442	0.6052
CLEANED	0.7344	0.5485	0.6978	0.6540	0.7416	0.6156	0.4637	0.4589	0.6143

Table 2: Prompt-by-prompt QWK results: *e.g.* `prompt 1` represents that prompt 1 is used as a test set, and prompts 2-7 as a training set, and prompt 8 as a dev set.

We present a comprehensive set of results utilizing various metrics in Table 3, including: Quadratic Weighted Kappa (QWK), Kendall Rank Coefficient (KRC), Spearman Rank Correlation (SRC), Pearson Correlation Coefficient (PCC), Mean Squared Error (MSE), and Root-Mean-Square Deviation or Root-Mean-Square Error (RMSD). This diverse range of metrics provides a thorough evaluation of our results from different perspectives.

	QWK	KRC	SRC	PCC	MSE	RMSD
ORIGINAL	0.6047	0.5477	0.6867	0.7062	0.0295	0.1676
ENCODING FIXED	0.6103	0.5426	0.6810	0.6982	0.0288	0.1652
CLEANED ALL'	0.6052	0.5479	0.6869	0.7050	0.0296	0.1675
CLEANED ALL	0.6143	0.5432	0.6813	0.7016	0.0287	0.1645

Table 3: Detailed results using various metrics.

[3] `https://huggingface.co/docs/transformers/main_classes/trainer`