

A Prompt Engineering Approach to Scientific Text Simplification: CYUT at SimpleText2023 Task3

Notebook for the CYUT Lab at CLEF 2023

Shih-Hung Wu^{*,†}, Hong-Yi Huang[†]

Chaoyang University of Technology, Taichung, Taiwan (R.O.C)

Abstract

This paper reports our approach to the SimpleText lab. In year 2023, we focus on the Task 3: Rewrite this!. Our system adopts the GPT3.5 and GPT4 generation models to rewrite the original sentences. We used different prompts to guide the model to generate simplified sentence with different guidelines. During system development, we used three metrics to evaluate the results. The official results are also reported.

Keywords

Simple Text Generation, Prompt Engineering, Evaluation of Text Simplification

1. Introduction

Understanding scientific texts requires proper background knowledge and academic terminology that makes the scientific texts hard to read. How to simplify the scientific text in an automatic way is the key point of the SimpleText lab.

The CLEF 2023 SimpleText lab is an evaluation campaign that aims to assess the quality and usability of text simplification systems. Text simplification is the task of rewriting a text in a simpler way, while preserving its meaning and information content. The lab will consist of three challenges of automatic text simplification in the following tasks:

- TASK 1: What is in (or out)? The goal of task 1 is given a query, a system has to find passages to include in a simplified summary.
- TASK 2: What is unclear? Given a passage and a query, a system has to rank terms that are required to be explained for understanding this passage.
- TASK 3: Rewrite this! Given a passage from scientific abstracts, a system has to rewrite it into a simplify passage.

SimpleText aims find the textual expression carrying information that should be simplified, the background information should be provided and the most relevant or helpful. Also system should try to improve the readability of a given short text. The lab provides a common framework

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

*Corresponding author.

[†]These authors contributed equally.

✉ shwu@cyut.edu.tw (S. Wu); s11027604@gm.cyut.edu.tw (H. Huang)

📞 0000-0002-1769-0613 (S. Wu); 0009-0005-6638-3029 (H. Huang)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

and dataset for comparing different approaches and measuring their impact on various aspects of text simplification, such as readability, comprehension, and preservation of meaning.

In this year, our team focus on Task3. We will describe our approach, how we evaluate our results and the official results in the following sections.

2. Techniques in Our Approach

The deep neural network that can generate natural language texts on various topics and domains. It is based on the transformer architecture, which uses attention mechanisms to learn the relationships between words and sentences. Transformer-based models have achieved state-of-the-art performance for abstractive summarization [1] [2] [3]. To our knowledge GPT4 [4] and T5 are the best ones. T5, or Text-to-Text Transfer Transformer [1], is a Transformer based architecture that uses a text-to-text approach. T5 can convert all NLP tasks in a Text-to-Text way with the help of prompt, a leading text to specify the goal that the user want this time. Usually, the prompts are trained tasks. The latest GPT models provide a more flexible way to give prompt in natural language, the use can give detail instruction on new tasks. Thus, how to design good prompt become an important issue on using the models [5] [6]. In SimpleText 2022, our system adopt the T5 model and get promising generation results [7]. The major drawback of the GPT4 model is it may not always produce factual or ethical texts, as it may inherit some biases or errors from the data it was trained on. The GPT4 model may also not be able to capture the nuances and contexts of human communication, such as sarcasm, humor, irony, etc. The GPT4 model may also require a lot of computational resources and energy to run and maintain. However, since it is a remarkable achievement in natural language processing and artificial intelligence, we explore its potential on the scientific text simplification.

2.1. Model Comparison

One of the main differences between T5 and GPT4 is their pre-training objectives. T5 is trained on a large corpus of text using a text-to-text framework, where it learns to map any input text to any output text. This allows T5 to perform a wide range of natural language tasks, such as summarization, translation, question answering, and text generation, by simply changing the output format. GPT4, on the other hand, is trained on a large corpus of text using an autoregressive language modeling objective, where it learns to predict the next word given the previous words. This allows GPT4 to generate fluent and coherent texts from a given prompt or context, but it also limits its ability to perform other natural language tasks that require more than word-level prediction.

Another difference between T5 and GPT4 is their model architectures. T5 is based on the Transformer encoder-decoder architecture, where it has two separate modules for encoding the input text and decoding the output text. This enables T5 to capture both the semantic and syntactic information from the input text and use it to generate the output text. GPT4 is based on the Transformer decoder-only architecture, where it has a single module for generating the output text from the input text. This simplifies the model design and reduces the computational cost, but it also makes it harder for GPT4 to incorporate external knowledge or information from the input text into the output text.

T5 and GPT4 are both powerful natural language processing models that can generate high-quality texts from various inputs. We tested both models on the dataset with our self-evaluation and find that GPT4 generate better results. Therefore, we use the GPT4 in formal test.

2.2. Prompt engineering

Prompt engineering is the process of carefully designing prompts that are provided to machine learning models, especially large language models like GPT-4, in order to guide their responses or behavior. The prompts go with the inputs to the model and the responses can vary greatly depending on how the prompts are crafted.

There are several techniques that can be used in prompt engineering, including but not limited to:

1. In-Context Learning [8]: Learning from the prompt and previous interactions to produce relevant responses.
2. Zero-Shot Prompting [9]: Providing a task unseen during training, testing the model's ability to use its learnt knowledge to respond.
3. Few-Shot Prompting [5]: Providing a few examples of the task before giving the actual prompt, helping the model understand the task.
4. Chain-of-Thought (CoT) Prompting [10]: Conditioning each prompt on the entire preceding conversation, maintaining context throughout a dialogue.

These techniques enable models to deliver more accurate and contextually relevant responses.

3. Datasets

SimpleText's data use the Citation Network Dataset: DBLP+Citation, ACM Citation network (12th version) as source of scientific documents to be simplified. Scientific textual content and authorship on any topic related to computer science can be extracted from this corpus. Detail description please read the overview paper [11]. The test set consist of two parts, we refer them as the large data set and the small data set. The large data set contains 152,072 records, and the small data set contains 2,234 records, 335KB.

4. Our Approach

In year 2023, we focus on Task3, here we give the detail of prompts used in our runs and the self-evaluation results.

4.1. Prompts Tested During System Development

Since the generation results can be very different under different prompts. We tried three prompts and two models listed in the following Table 1. It is an essential process to use natural language models effectively and responsibly. We manually ask GPT4 model to improve the original prompt, (simplify the text), and get these prompts. Our system put every test sentence into the text slot and call the API provided by OpenAI. As we can see, GPT4 suggest that prompt

Table 1
Prompts used in different runs.

Run	Model/Dataset	Prompt
1	GPT-3.5/large data	Your task is to simplify the following sentences to make them easier to understand. Please note that your response should be flexible enough to allow for various relevant and creative simplifications, as long as they accurately convey the intended meaning. Please simplify this sentence:{text}
2	GPT-4/small data	
3	GPT-4/small data	Your task is to simplify the following sentences to make them easier to understand. Please provide a clear and concise response that retains the original meaning of each sentence while removing any unnecessary complexity or jargon. Please note that your response should be flexible enough to allow for various relevant and creative simplifications, as long as they accurately convey the intended meaning. Please simplify this sentence:{text}
4	GPT-4/small data	Simplify these sentences to make them easier to understand while retaining their meaning and avoiding complex language. Be creative in your simplification. Please simplify this sentence:{text}

should give detailed instruction, such as “easier to understand”, “various relevant”, “creative simplification”, “removing ... jargon”, and “retaining ... meaning”.

4.2. Self-Evaluation

Before sending the **generated text** to the organizers, we **evaluate them with three metrics**. Table 2 gives the evaluation results of the source dataset and our runs. **Flesch reading ease is an index that measures the level of sentences; the higher the score, the easier it is for the reader [12]**. In Table 2, we can see that the reading ease index value for run 2 is the highest. **The Flesch-Kincaid grade level (FKGL) is an index that measures the corresponding reader level; the lower the score, the younger the reader [13]**. This is a grade index where a score of 10.x means that a tenth-grader would be able to read the text. Run 2 also has the lowest level, which drops from 14.61 to 9.59. The evaluation results in Table 2 show that, with the same model and input data, the prompt affects the results significantly. In the last column of Table 2, we can find the percentage of academic words according to a list provided by Coxhead [14]. The percentage of academic words drops to 9.65% in run 2.

Table 3 shows a generation results example in our evaluation, where FRE is the Flesch Reading Ease, FKGL is the Flesch-Kincaid Grade Level, and length is the number of characters in the sentence. We can see the increase of FRE and decrease of the FKGL and the shorten of the length in all four runs. The generated sentences preserve the meaning of the source sentence in this case. The country name, Greece, in the source sentence is replaced by the adjective, Greek, in all the three GPT4 generated sentences. The last one is a little bit creative, it added “improving

Table 2

Flesch-Kincaid readability and the Academic Word Profile in runs.

Run	Model/Dataset	The Flesch Reading Ease	The Flesch-Kincaid Grade Level	Percentage of academic words
-	source/large data	28.04	14.78	15.64%
-	source/small data	29.2	14.61	15.26%
1	GPT-3.5/large data	49.46	11.0	11.42%
2	GPT-4/small data	58.11	9.59	9.65%
3	GPT-4/small data	49.04	10.95	11.84%
4	GPT-4/small data	49.76	10.72	11.95%

Table 3

A Typical Example in our evaluation

Run	Sentence	FRE	FKGL	length
source	The application is to be used by firefighting personnel in Greece and is potentially expected to contribute towards a more sophisticated transferring of information and knowledge between wildfire confrontation operation centers and firefighting units in the field.	8.54	21.3	264
1	This app will be used by firefighters in Greece to improve the sharing of information and knowledge between wildfire response centers and those working in the field.	52.53	12.6	165
2	The app helps Greek firefighters share information and knowledge between wildfire centers and field teams more easily.	54.22	9.9	118
3	The app helps Greek firefighters share important information and knowledge between operation centers and units fighting wildfires.	28.84	13.5	130
4	The app will help Greek firefighters share information and knowledge between operation centers and field units, improving wildfire response.	35.27	13.1	140

*FRE: Flesch reading ease

*FKGL: Flesch-Kincaid grade level

wildfire response”, which is not in the source sentence.

5. Official Results

We participated in the SimpleText challenge under the name "CYUT". The official evaluation result of our runs in the Task3 is listed in the following tables. Where the Max, Min, and Avg are the maximum value, the minimum value, and the average values of all runs in SimpleText2023. The major metrics are FKGL and SARI, two public available automatic evaluation metrics, which

Table 4

Flesch-Kincaid Grade Level (the lower the better)

runs	task3 test	task3 train
CYUT_task_3_run1	9.63	10.22
CYUT_task_3_run2	8.43	9.18
CYUT_task_3_run3	10.00	10.48
CYUT_task_3_run4	9.24	10.29
Max	13.05	14.63
Min	7.53	8.08
Avg	11.29	11.78

Table 5

SARI (0-100) (the higher the better)

runs	task3 test	task3 train
CYUT_task_3_run1	47.98	35.74
CYUT_task_3_run2	44.93	34.71
CYUT_task_3_run3	46.82	36.24
CYUT_task_3_run4	47.70	36.59
Max	47.98	89.95
Min	23.28	27.53
Avg	35.25	43.53

are not used in the evaluation last year. We also used FKGL at self-evaluation while developing our system, but we never use SARI before.

SARI is a metric that evaluates how well automatic text simplification systems rewrite sentences to make them easier to read and understand [15]. SARI compares the predicted simplified sentences with the original sentences and the human references. SARI calculates the quality of the words that are added, deleted, or kept by the system, based on how they match the human references. SARI is a general metric that can capture the effects of different simplification operations, such as lexical paraphrasing, syntactic restructuring, or information deletion. This year, our first run gives the highest SARI score 47.98 among all participant runs. The corresponding FKGL levels are around 9 grade and the lexical complexity scores are also at the lowest level.

6. Conclusion and Discussions

In terms of generating sentences for Task 3, the results are much improved from the results in last year. We observed the excess parts of the sentence removed, the academic terminology is replaced with common words and finally the simplified sentence is obtained. The Simplified sentence fully express the meaning of the original sentence. The GPT4 model gives better results on task 3 than T5 model with the help of prompt engineering [5].

Text simplification is the process of transforming a complex text into a simpler one, while

Table 6

Lexical complexity score (the lower the better)

runs	task3 test	task3 train
CYUT_task_3_run1	8.35	8.40
CYUT_task_3_run2	8.31	8.26
CYUT_task_3_run3	8.36	8.29
CYUT_task_3_run4	8.33	8.32
Max	8.68	8.79
Min	8.27	8.26
Avg	8.49	8.63

preserving its meaning and information content. Current deep neural network models can give better results than old ones. However, evaluating the quality of simplified texts is getting difficult.

There are different aspects of simplicity, such as lexical, syntactic, semantic, and pragmatic. Lexical simplicity refers to the use of common and familiar words, syntactic simplicity refers to the use of short and simple sentences, semantic simplicity refers to the use of clear and unambiguous meanings, and pragmatic simplicity refers to the use of appropriate and relevant information for the intended audience. However, these aspects are not independent and may interact with each other in complex ways. To balance simplicity with other quality criteria, such as adequacy, coherence, and informativeness, may cause confliction between metrics.

Automatic metrics, such as readability formulas, lexical diversity measures, or compression ratios, can provide objective and quantitative scores for some aspects of simplicity and quality, may not capture all the nuances and subtleties of human language. Human judgments, such as ratings, rankings, or preferences, can provide subjective and qualitative feedback for various aspects of simplicity and quality, costly but still necessary.

Acknowledgments

This study was supported by the National Science and Technology Council under the grant number NSTC 112-2221-E-324-014.

References

- [1] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, *The Journal of Machine Learning Research* 21 (2020) 5485–5551.
- [2] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint arXiv:1810.04805* (2018).
- [3] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, *arXiv preprint arXiv:1910.13461* (2019).

- [4] OpenAI, Gpt-4 technical report, 2023. [arXiv:2303.08774](https://arxiv.org/abs/2303.08774).
- [5] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, *Advances in neural information processing systems* 33 (2020) 1877–1901.
- [6] J. Huang, S. S. Gu, L. Hou, Y. Wu, X. Wang, H. Yu, J. Han, Large language models can self-improve, *arXiv preprint arXiv:2210.11610* (2022).
- [7] S.-H. Wu, H.-Y. Huang, Cyut team2 simpletext shared task report in clef-2022, *Proceedings of the Working Notes of CLEF* (2022).
- [8] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, Z. Sui, A survey for in-context learning, *arXiv preprint arXiv:2301.00234* (2022).
- [9] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, Q. V. Le, Finetuned language models are zero-shot learners, *arXiv preprint arXiv:2109.01652* (2021).
- [10] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al., Chain-of-thought prompting elicits reasoning in large language models, *Advances in Neural Information Processing Systems* 35 (2022) 24824–24837.
- [11] L. Ermakova, E. SanJuan, S. Huet, O. Augereau, H. Azarbonyad, J. Kamps, Overview of simpletext - clef-2023 track on automatic simplification of scientific texts, in: E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. lielmo Faggioli, N. Ferro (Eds.), *Avi Arampatzis, Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023)*, 2023.
- [12] R. Flesch, *How to write plain english: Let's start with the formula*, University of Canterbury (1979).
- [13] J. P. Kincaid, R. P. Fishburne Jr, R. L. Rogers, B. S. Chissom, Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel (1975).
- [14] A. Coxhead, A new academic word list, *TESOL quarterly* 34 (2000) 213–238.
- [15] W. Xu, C. Napoles, E. Pavlick, Q. Chen, C. Callison-Burch, Optimizing statistical machine translation for text simplification, *Transactions of the Association for Computational Linguistics* 4 (2016) 401–415.