A Theoretical Approach for Automatic Textual Data Processing Incorporating Prompt Engineering

Jinmo Yang, Janghwan Kim, Chaeyun Seo, and R. Young Chul Kim*
SE Lab, Hongik University
Sejong, South Korea
[e-mail: yjmd2222@g.hongik.ac.kr, {lentoconstante, chaeyun, *bob}@hongik.ac.kr]
*Corresponding author: R. Young Chul Kim

Abstract

Nowadays, large language models (LLMs) may be an absolute necessity for the highest productivity and maximum gain in all business domains for their power of idea generation, business plans, etc. Having learned vast and diverse information from tremendous amounts of textual sentences, LLMs are generating large quantities of results in rich quality. Current problems with LLMs are paying fees for LLM services and manual modifications for high quality data. To solve these problems, we propose an automatic textual data process that incorporates prompt engineering with LLMs that require no fees. We assess the plausibility of our process with Korean Language Understanding Evaluation (KLUE) benchmark dataset for the textual dataset. Our future work includes assessing generalization of this process for other LLMs, fully applying our process on KLUE dataset for data augmentation and using the augmented dataset to fine-tune LLMs for their higher Korean proficiency.

***Keywords*:** data augmentation, natural language processing, prompt engineering, tokens

1. Introduction

Since the release of sensational ChatGPT3 [1] in 2022, large language models (LLMs) are gaining more attention than ever. From simple tasks as answering questions to advanced ones as understanding user contexts to help create business items, LLMs are acting as employees for business domains for maximum gain [2]. And with AI tools such as OpenAI's Playground, users can customize the various settings, prompt-engineer the user-AI interactions, and maximize the potential of the models. However, there is pricing on the OpenAI API usage, where charges are applied for each token in user input and LLM output. This is a major concern in all business as they focus on providing personalized user experience from all kinds of user data [3]. Additionally, the collected user data often needs manual preprocessing before it can be used [4].

On a different note, the structure of most LLM APIs is based on the Transformers library in Hugging Face [5]. The Transformers library provides many LLMs and commands that can be used on local computers, given that the computer specifications suffice. Now, this library itself is free of cost as the processing of the input for various tasks are done on the local computers.

Given the abovementioned, we propose an automatic textual data process that incorporates prompt engineering with LLMs. Using the Transformers library for free-of-charge interface, per-use charging of widely used APIs can be prevented. Also, using prompt-engineering techniques, data can be stored in a consistent format. As an example, we show the process of augmenting a large benchmark dataset of Korean news headlines from Korean Language Understanding Evaluation (KLUE) into formal and colloquial sentences. The rest of the paper is as follows. Section 2 mentions background research. Section 3 presents our process. Section 4 shows the example with KLUE data. And finally, Section 5 mentions our conclusion.

2. Background Research

2.1 Prompt Engineering

LLMs are notable for their capabilities of understanding input and producing output a user requires. There are many prompt-engineering techniques to enhance the quality of the interaction with and the output from the LLMs, such as zero-shot, chain-of-thought, reflexion, and prompt chaining [6]. For example, prompt chaining is a technique for asking a series of questions, where the output of the preceding question is used as a part of the input in the successive question. This way, the LLM brainstorms at each smaller sub-task instead of trying to answer one whole question in a single step. This helps in debugging the reasoning of the LLM. And when all sub-tasks are done, the LLM can have a clear understanding of the context, effectively gaining the likelihood of generating a high-quality output [7].

2.2 OpenAI API

Table 1. OpenAI pricing for selected APIs

API	Pricing per 1M tokens
gpt-4o	$ 2.50 / input
	$ 10.00 / output
Complex tasks o1-preview	$ 15.00 / input
	$ 60.00 / output
Fine-tuning gpt-4o-2024-08-06	$ 3.750 / input
	$ 15.000 / output
	$ 25.000 / training

OpenAI API provides numerous features for productive usage of AIs. Key features include retrieving knowledge from the web and data, running code and fine-tuning models in a cloud environment, and answering questions about images and videos [8]. In exchange for using these powerful features, charges are applied on a token basis.

Table 1 shows the pricing for selected APIs. To process data in certain quantities, e.g. entering inputs and returning outputs for over 300 customers in a GPT application service, the corresponding fee of $ 20 per day must be made [9]. If the user count increased to hundreds of millions, such as Netflix's over 260 million subscribers as of 2023, a fee of $ 17 M would be charged per day [10].

2.3 LLM-Driven Workflows

There is recent application research on LLM-incorporated data processing. For example, Zhou *et al.* [11] proposed the LLM-enhanced data management system, where LLMs were used throughout the management process. Although the process is well-defined, the preparation for building the system is overwhelming with too many tools and LLMs which would require high memory and multiple GPU usage.

Patel *et al.* [12] proposed DataDreamer, an open source library aimed at providing a workspace that is easy to work in. As a use case, they created a synthetic data generation workflow for data augmentation. Whereas useful features are provided with this library, it may just be a wrapper for existing APIs.

3. Automatic Textual Data Processing Incorporating Prompt Engineering

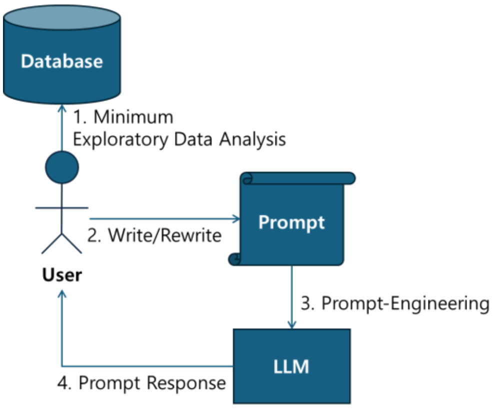


Fig. 1. Prompt-Engineering

The automatic textual data processing uses prompt-engineering-driven LLM to process

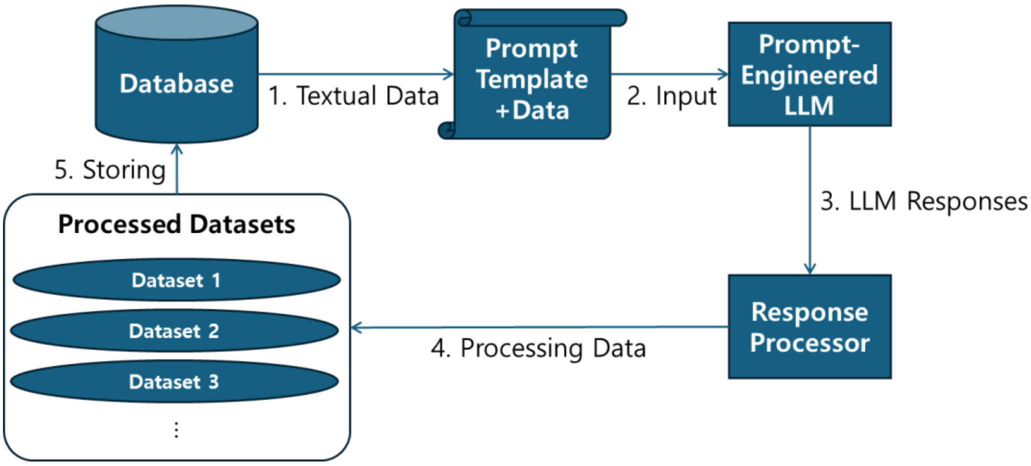


Fig. 2. Automatic Textual Processing

textual data. This can be best understood in two parts: prompt-engineering and automatic textual processing.

Fig. 1 shows the schematics of prompt-engineering. In Step 1, a user performs minimum exploratory data analysis on the data. This is to ensure that the user understands the data to write prompts in the appropriate context. In Step 2, the user writes a prompt for the LLM with understanding from the data. In Step 3, the prompt is applied to the LLM, with various prompt-engineering techniques where they may apply. In Step 4, the user receives a prompt response from LLM. To fully teach the LLM on data processing, Steps 2-4 are repeated as necessary.

Fig. 2 shows automatic textual processing after the user has completed prompt-engineering. The engineered LLM is integrated into the automatic textual process as Prompt-Engineered LLM. In Step 1, raw textual data is extracted from the database and combined into a predefined prompt template. This template is to help the LLM process the data and assess the output for validity. In Step 2, the prompt template with data is entered into the LLM as the input. In Step 3, LLM generates responses and passed to the Response Processor. In Step 4, the Response Processor processes data into datasets. And in Step 5, the processed datasets are stored into the database.

Having learned the data-related context from prompt-engineerig, the engineered LLM is able to process input data and produce output that the user expects.

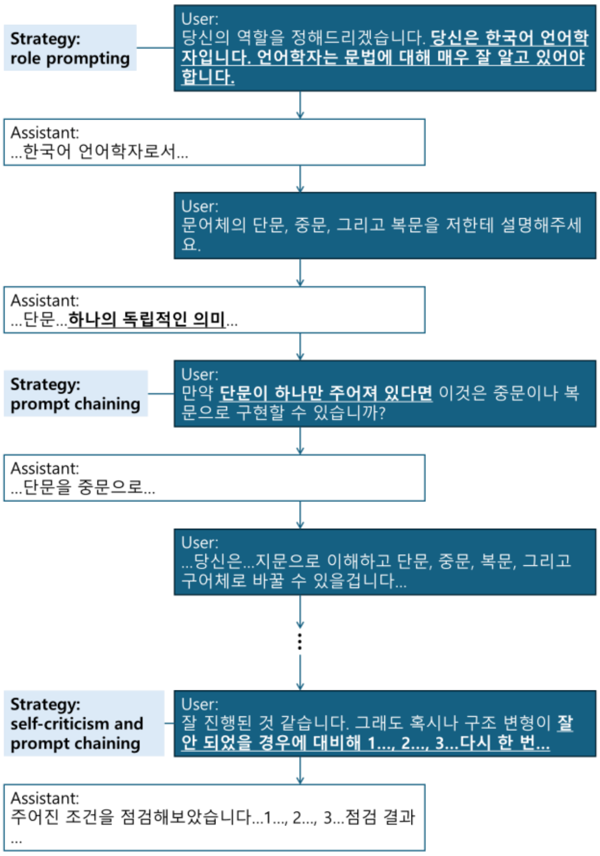
4. Applying the Process for Korean Language Understanding Evaluation Dataset Augmentation

To test the plausibility of our proposed process, we prompt-engineer ChatGPT3-4o to understand Korean grammar, specifically sentence types, and look at the output sentences. Fig. 3 is a shortened dialogue between a user and the AI.

The conversation between the user and ChatGPT is about how to generate formal and colloquial sentences from a given passage. The prompting techniques used were role prompting, prompt chaining, and self-criticism [6].

Role prompting was used to make the AI act as a Korean linguist. This forces the LLM to restrict its knowledge domain to Korean languages thereby having it focus on Korean linguistics and not confuse from information in interdomains.

Prompt chaining was used so that the AI would think in between prompts and that problematic future responses are reduced by refining previous prompts. The AI thought about different types of formal and colloquial sentences before generating them from a given passage. The underlined phrase “단문이 하나만 주어져 있다면” (which translates to “if just one simple sentence was given”) was added because the AI generated a response where a single simple sentence was transformed into compound and complex sentences with context outside of the



User:
당신의 역할을 정해드리겠습니다. **당신은 한국어 언어학자입니다. 언어학자는 문법에 대해 매우 잘 알고 있어야 합니다.**

Assistant:
...한국어 언어학자로서...

User:
문어체의 단문, 중문, 그리고 복문을 저한테 설명해주세요.

Assistant:
...단문...**하나의 독립적인 의미**...

User:
만약 **단문이 하나만 주어져 있다면** 이것은 중문이나 복문으로 구현할 수 있습니까?

Assistant:
...단문을 중문으로...

User:
...당신은...지문으로 이해하고 단문, 중문, 복문, 그리고 구어체로 바꿀 수 있을겁니다...

User:
잘 진행된 것 같습니다. 그래도 혹시나 구조 변형이 **잘 안 되었을 경우에 대비해 1..., 2..., 3...다시 한 번...**

Assistant:
주어진 조건을 점검해보았습니다...1..., 2..., 3...점검 결과
...

Fig. 3. Prompt-engineering ChatGPT with instructions, truncated

passage, e.g. "모니터 받침대를 사용할 필요가 없어서, 책을 쌓아서 사용하고 있습니다" from "모니터 받침대 대신 책 쌓아서 쓰는 중" (The original passage means "Currently using a pile of books instead of a monitor stand," and the transformed complex sentence means "Because I have no need for a monitor stand, I am using a pile of books instead.").

Self-criticism was used so that the AI would assess its own answer for legitimacy. The initial response may have mixed mapping among sentence types and introduction of outside context. One thing to note is that the prompts using this technique were written in neutral expressions so that the AI assesses its own responses for better ones only if they were initially incorrect and that the AI does not alter its correct responses from thinking that the user request suggests the responses were incorrect.

When given a few test and real datapoints, the engineered ChatGPT generated correct outputs. **Table 2** shows comparison between the outputs from the engineered AI and the non-engineered AI for a chosen passage.

The input passage was "유튜브 내달 2일까지 크리에이터 지원 공간 운영" (which translates to "Youtube runs a creator support space until the second day of the next month"). While the majority of the transformed sentences were correct, the compound and complex sentences in non-engineered outputs were incorrect (written in red). These sentences could not have been generated from a simple passage, but the AI marked them as correctly generated. In addition, outside context was added to the complex sentence, i.e. "크리에이터들이 편하게 사용할 수 있도록" (which translates to "for the creators to comfortably use") which the original passage did not explicitly mention. Although not shown in the table, all final outputs from the engineered ChatGPT meet the user requirement. This shows that the prompt-engineered ChatGPT can be used to process large data in a consistent format.

Table 2. Prompt-engineered vs. non-engineered ChatGPT output

Original passage	Engineered	Transformations	translation_done (AI)	translation_done (user)	Outside context added	Appropriate
유튜브 내달 2일까지 크리에이터 지원 공간 운영	Yes	유튜브는 내달 2일까지 크리에이터 지원 공간을 운영한다.	True	True	False	Yes
		유튜브는 내달 2일까지 크리에이터 지원 공간을 운영한다.	False	False	False	Yes
		유튜브가 내달 2일까지 크리에이터 지원 공간을 운영한다.	False	False	False	Yes
		유튜브가 내달 2일까지 크리에이터 지원 공간 운영한대.	True	True	False	Yes
	No	유튜브가 내달 2일까지 크리에이터 지원 공간을 운영합니다.	True	True	False	Yes
		유튜브는 크리에이터 지원 공간을 내달 2일까지 운영할 계획입니다.	True	False	False	No
		유튜브는 크리에이터들이 편하게 사용할 수 있도록 지원 공간을 마련해 내달 2일까지 운영할 예정입니다.	True	False	True	No
		유튜브가 다음 달 2일까지 크리에이터 공간을 운영한다고 하네요.	True	True	False	Yes

Parsing the prompt-engineered test outputs with the response processor, the resulting json data was correctly committed to a NoSQL database. Therefore, if we transfer the prompts to the Hugging Face interface, our proposed process can process large amounts of data without any pricing on the token usage without manual modifications.

The chat used in this section can be found at https://chatgpt.com/share/6722490f-ba38-8010-94af-800526a0ca6f.

5. Conclusion

We mentioned an automatic textual data processing incorporating prompt engineering. As an example, we have presented an application scheme for KLUE data augmentation with the proposed process and conducted basic tests where the data processing from the LLM was correctly done and the result was committed to the database. Our process can be used to process large datasets without charges and manual modifications. Challenges remain for applying effective prompt-engineering techniques and conducting minimum exploratory data analysis on the data to be automatically processed. For future work, we will generalize this process with LLMs with the Transformers interface, fully apply the textual process with KLUE data, and fine-tune LLMs with this processed data for higher Korean language proficiency.

Acknowledgment

This research was supported by Korea Creative Content Agency (KOCCA) grant funded by the Ministry of Culture, Sports and Tourism (MCST) in 2024 (Project Name: Artificial Intelligence-based User Interactive Storytelling 3D Scene Authoring Technology Development, Project Number: RS-2023-0022791730782087050201) and National Research Foundation (NRF), Korea, under project BK21 Four.

References

[1] OpenAI, Introducing ChatGPT, OpenAI, https://openai.com/index/chatgpt/, Accessed October 31, 2024.

[2] M. Abdullah, A. Madain, and Y. Jararweh, “ChatGPT: Fundamentals, Applications and Social Impacts,” in *Proc. of 2022 Ninth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, 2022.

[3] M. Raji, H. Olodo, T. Oke, W. Addy, O. Ofodile, and A. Oyewole, “E-commerce

and consumer behavior: A review of AI-powered personalization and market trends," *GSC Advanced Research and Reviews*, vol.18, No.3, pp.66-77, 2024.

[4] G. Cha, "Gender-neutral employment from AI, generative AIs for Serious Accidents Punishment Act, and newly emerging privacy issues in the generative AI paradigm," (invited lecture), *2024 Autumn Academic Conference of Smart Media*.

[5] T. Wolf et al., "HuggingFace's Transformers: State-of-the-art Natural Language Processing," *arXiv*, 2019.

[6] S. Schulhoff et al., "The Prompt Report: A Systematic Survey of Prompting Techniques," *arXiv*, 2024.

[7] T. Wu, M. Terry, and C. Cai, "AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts," in *Proc. of 2022 CHI Conference on Human Factors in Computing Systems*, no.385, pp.1-22, 2022.

[8] OpenAI, The most powerful platform for building AI products, OpenAI, https://openai.com/api/, Accessed October 31, 2024.

[9] ckkx4hdzkd, SOS: ALARMING Situation of Excessive Billing Threatening the Survival of my Company AI Project GPT, OpenAI community, https://community.openai.com/t/sos-alarming-situation-of-excessive-billing-threatening-the-survival-of-my-company-ai-project-gpt/734483/1, Accessed October 31, 2024.

[10] Netflix, Inc., 2023 Annual Report, Netflix Inc, 2023.

[11] X. Zhou, X. Zhao, and G. Li, "LLM-Enhanced Data Management," *arXiv*, 2024.

[12] A. Patel, C. Raffel, and C. Callison-Burch, "DataDreamer:A Tool for Synthetic Data Generation and Reproducible LLM Workflows," *arXiv*, 2024.