SELF-EXPERTISE: Knowledge-based Instruction Dataset Augmentation for a Legal Expert Language Model

Minju Kim* **Haein Jung*** **Myoung-wan Koo†**

Department of Artificial Intelligence, Sogang University, Republic of Korea
{mjmjkk0307, haeindain, mwkoo}@sogang.ac.kr

Abstract

The advent of instruction-tuned large language models (LLMs) has significantly advanced the field of automatic instruction dataset augmentation. However, the method of generating instructions and outputs from inherent knowledge of LLM can unintentionally produce hallucinations — instances of generating factually incorrect or misleading information. To overcome this, we propose SELF-EXPERTISE, automatically generating instruction dataset in the legal domain from a seed dataset. SELF-EXPERTISE extracts knowledge from the outputs of the seed dataset, and generates new instructions, inputs, and outputs. In this way, the proposed method reduces hallucination in automatic instruction augmentation. We trained an SELF-EXPERTISE augmented instruction dataset on the LLaMA-2 7B model to construct Korean legal specialized model, called LxPERT. LxPERT has demonstrated performance surpassing GPT-3.5-turbo in both in-domain and out-of-domain datasets. The SELF-EXPERTISE augmentation pipeline is not only applicable to the legal field but is also expected to be extendable to various domains, potentially advancing domain-specialized LLMs.

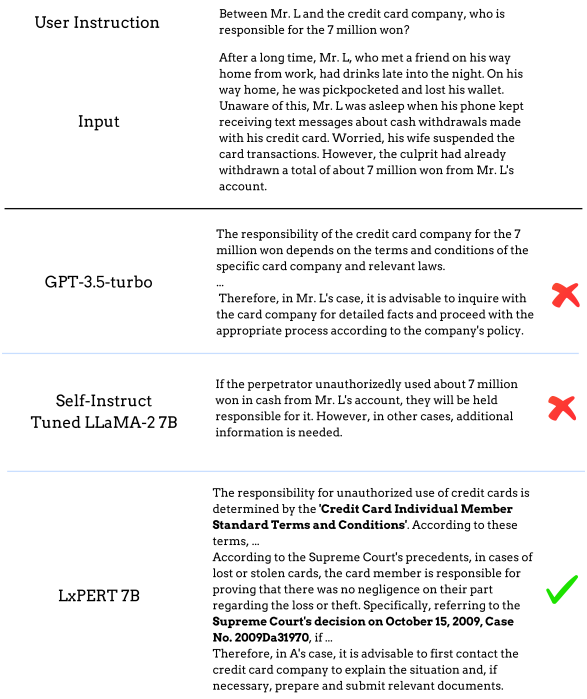

Figure 1: Comparison between GPT-3.5-turbo, Self-Instruct tuned LLaMA-2, and our LxPERT. GPT-3.5-turbo and Self-Instruct tuned LLaMA-2 generates unclear and ambiguous output without any legal basis. By contrast, LxPERT generates correct output with precise legal knowledge.

1 Introduction

Recent large language models (LLMs) like GPTs (Brown et al., 2020; OpenAI, 2023), PaLM (Chowdhery et al., 2023) show exceptional performance for various NLP tasks in response to instruction prompt. To train LLMs in following natural language instructions, an instruction dataset is essential for performing Instruction Tuning (Yin et al., 2023; Wei et al., 2021; Sanh et al., 2021). However, acquiring a diverse human-written instruction dataset is often time-consuming and costly. To overcome these limitations, many have used LLMs for auto-generation (Wang et al., 2022; Xu et al., 2023). However, such methods remain unsuitable for creating instruction datasets in domains where accuracy is critical, because LLMs lack expert-level knowledge, can produce hallucinated data (Lin et al., 2021; Zhang et al., 2023; Sadat et al., 2023). Therefore, high-quality, accuracy-focused methodologies are needed for generating instruction data in specialized knowledge areas to extend the usability of LLMs beyond general tasks.

In this study, we introduce the novel **SELF-EXPERTISE** method for automatic instruction data generation for knowledge-intensive tasks using LLMs. It prevents hallucinations and ensuring data accuracy in specialized knowledge do-

*These authors contributed equally to this work.
†Corresponding author

Findings of the Association for Computational Linguistics: NAACL 2024, pages 1098–1112

mains by generating data based on precise knowledge. Figure 2 describes four-step process of SELF-EXPERTISE. Inspired by how a teacher designs exam questions based on textbook context, user instructions, inputs, and outputs are created based on the knowledge, extracted from small set of seed data outputs.

We automatically generated a legal domain instruction dataset of 19k from 980 seed dataset utilizing SELF-EXPERTISE. Then, we instruction tuned LLaMA-2 7B (Touvron et al., 2023) using SELF-EXPERTISE augmented dataset. We refer to this resulting model as LxPERT, for Legal ExPERT. Comparisons of LxPERT with models instruction-tuned in general domains and those tuned with datasets generated by traditional augmentation methods reveal its superior accuracy and fluency. Furthermore, LxPERT significantly surpasses GPT-3.5-turbo, the most widely used model lately. Figure 1 shows comparison between GPT-3.5-turbo and LxPERT. These findings underscore the importance of specialized models for specific domains (Zhao et al., 2023; Chalkidis et al., 2020; Tian et al., 2023), highlighting the effectiveness of the SELF-EXPERTISE approach in developing models that deliver high performance in professional fields. In summary, our contributions are as follows:

- We propose SELF-EXPERTISE, a novel instruction data generation method for areas of specialized knowledge that minimizes human annotation.
- We train LxPERT, the small large language model (sLLM) specialized in Korean legal domain using SELF-EXPERTISE method. We conduct both GPT-4 and human evaluation on in-domain and out-of-domain test set. The results demonstrate that LxPERT surpasses the 7B models and GPT-3.5-turbo with high accuracy.
- We release a Korean Legal SELF-EXPERTISE Instruction Dataset and a set of handcrafted novel dataset for evaluating future legal instruction-following models.

2 Related Work

2.1 LLM-based Instruction Dataset Augmentation

Collecting diverse instruction datasets manually requires significant resources (Ratner et al., 2017; Zhong et al., 2020; Feng et al., 2021). To overcome these limitations, methods have been proposed to automatically generate instruction datasets through LLMs (Dai et al., 2023; Whitehouse et al., 2023; Wang et al., 2022; Xu et al., 2023; Mukherjee et al., 2023; Mitra et al., 2023). Self-Instruct (Wang et al., 2022) presents a method to generate instructions by looking up few task examples from a seed dataset, produce outputs, and then filter out low-quality data. However, in specialized knowledge domains, including law, where accurate answers based on expert knowledge are required, there are limitations to controlling LLM hallucination when using previous automatic generation methods (Choi et al., 2023; Yu et al., 2023a; Cui et al., 2023). When a new instruction is created without incorporating related knowledge, LLMs may generate inaccurate answer. SELF-EXPERTISE overcomes these limitations by extracting knowledge from the output of existing seed dataset and generating new dataset based on this knowledge. It also incorporates explanation tuning (Mukherjee et al., 2023) to enable learning a logical answer structure.

2.2 Knowledge-Intensive Tasks

Knowledge-intensive tasks require a knowledge-based solution, such as open domain question answering, fact-checking, and entity linking (Petroni et al., 2020). The legal domain is knowledge-intensive because answers must be provided based on accurate information (Yu et al., 2023b; Kim and Goebel, 2017; Vold and Conrad, 2021). While LLMs have shown high performance in knowledge-intensive tasks using only model parameters, they still face the limitation of hallucination (Asai et al., 2023; Lewis et al., 2020; Guu et al., 2020). This problem can be reduced when accurate knowledge is added to the LLM input through a retriever (Shuster et al., 2021; Borgeaud et al., 2022; Mallen et al., 2023; Shi et al., 2023). Accordingly, we apply a similar method to data augmentation for knowledge-intensive tasks. Unlike previous instruction dataset generation methods that generates instructions from undefined inherent knowledge of LLMs, our method creates instructions and outputs based on precise external knowledge.

3 Methodology

We propose SELF-EXPERTISE, a novel methodology for automatically generating instruction data based on knowledge, thus enabling precise logical reasoning that reflects the characteristics of special-

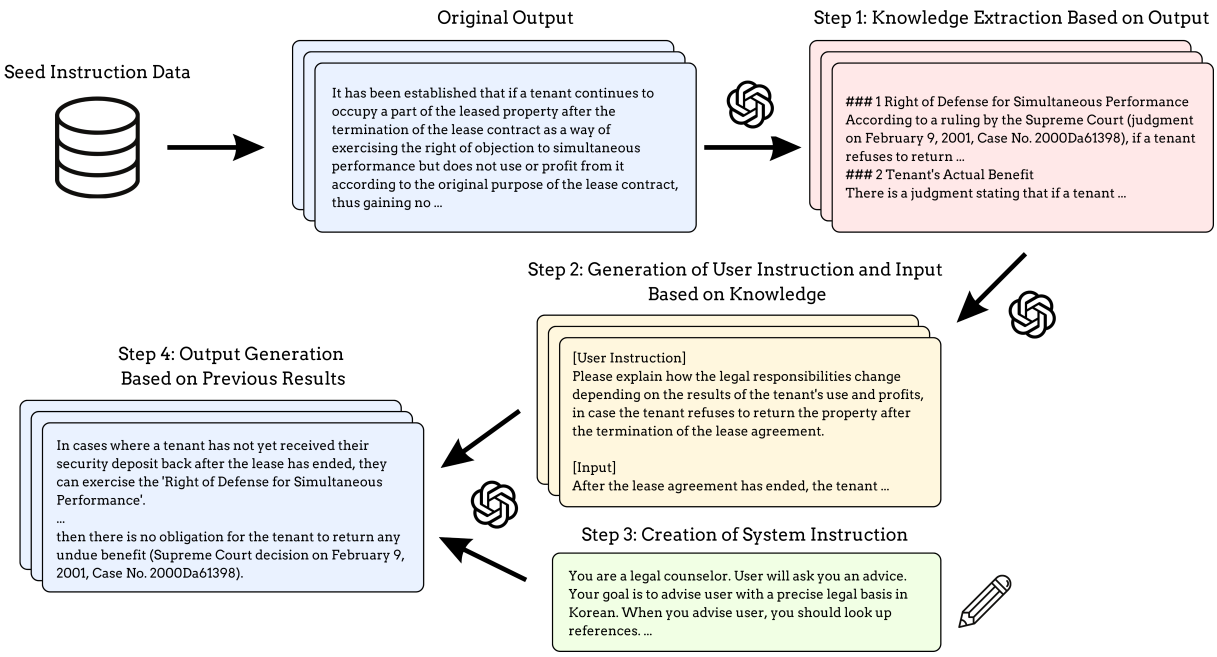

Figure 2: A overview of SELF-EXPERTISE.

ized knowledge areas.

3.1 Defining Instruction Data

A typical instruction dataset (i.e., seed dataset) is structured as <user instruction, input, output> (Wei et al., 2021). It is designed to generate an output from the model when a user instruction and corresponding input are provided. In our methodology, we add system instructions (Mukherjee et al., 2023). Unlike user instructions that direct the actual task to be performed by the model, system instructions serve as guidelines for additional details such as the tone or style of the output. While focusing on the importance of logical structure in specialized knowledge domain responses, we design and add system instructions to facilitate learning of reasoning and narrative structure. The final dataset is structured as <system instruction, user instruction, input, output>. Both cases, with and without inputs, are structured for a diverse instruction dataset.

3.2 SELF-EXPERTISE

SELF-EXPERTISE involves four stages: (1) knowledge extraction based on output, (2) generation of user instruction and input based on knowledge, (3) creation of system instructions, and (4) output generation based on previous results. (Figure 2) The prompt template used in each step is shown in Figure 3.

3.2.1 Step 1: Knowledge Extraction Based on Output

First, knowledge is extracted from the outputs of a small set of expert-written seed data. Unlike the conventional method (Wang et al., 2022) that generates new user instructions and outputs solely based on inherent knowledge of LLM, our method generates user instruction, input, and output based on precise external knowledge. This is crucial in specialized knowledge areas where factual accuracy matters. Thus, using accurate knowledge as a basis for data generation can prevent hallucination and ensure data accuracy. For example, in the legal field, a lawyer's argument corresponds to the output, and the case law used as the basis for the argument corresponds to the knowledge.

3.2.2 Step 2: Generation of User Instruction and Input Based on Knowledge

User instruction and input are generated using the extracted knowledge from the previous step. Analogous to how teachers create exam questions based on textbook content, LLM acts as an exam writer and generates relevant exam questions and contexts (i.e., user instruction and input) based on knowledge.

3.2.3 Step 3: Creation of System Instructions

To generate diverse outputs utilizing knowledge, specialized system instructions are handcrafted as

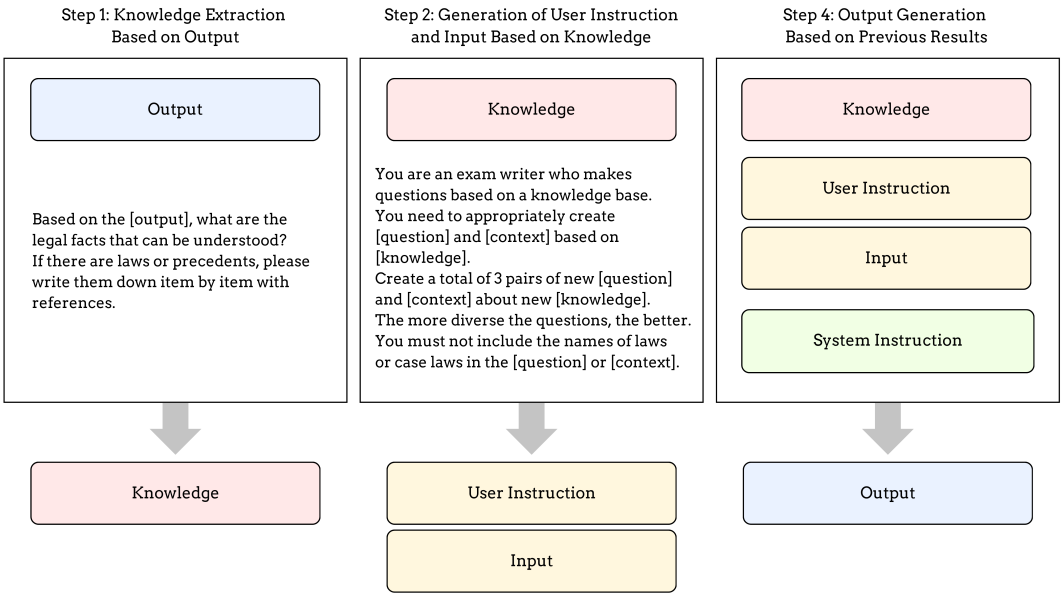

Figure 3: Prompt templates used for each step during SELF-EXPERTISE.

guidelines for output generation. In this study, eight specific system instructions were handcrafted for the targeted legal field. All system instructions include the use of precise legal basis and instruction to generate output referencing the knowledge. Individually, they differ to allow the creation of outputs in various manners, lengths, and formats. For example, one follows the stages of the Issue, Rule, Application, Conclusion(IRAC) Framework, a real legal reasoning process, to align outputs with the thought process in specialized knowledge areas.

3.2.4 Step 4: Output Generation Based on Previous Results

Finally, the output is generated using the system instruction, user instruction, input, and knowledge. Upon combining the previously generated user instruction, input, and knowledge with the eight system instructions, eight outputs are generated for each user instruction and input pair. During the output generation process with LLM, accurate knowledge is included in the prompt to ensure the accuracy of the output.

3.3 Finetuning the sLLM using Augmented Instruction Dataset

The small LLM (sLLM) is trained in causal language modeling using augmented instruction dataset. This process can be seen as knowledge distillation, in which knowledge is transferred from a larger to a smaller model (Wang et al., 2022). Like previous studies, our approach not only transfers the knowledge and instruction-following ability of the larger model but also allows for the distillation of domain knowledge in specialized fields. Unlike generating data with LLM, where knowledge is included to generate outputs, knowledge is not directly provided during the training of the sLLM. Therefore, the sLLM is trained to generate responses based on indirectly learned domain knowledge when receiving instruction and input, considering that in real user query scenarios, ground-truth knowledge rarely comes as input. Moreover, the sLLM learn all eight types of system instructions and their corresponding various output forms. This allows the sLLM to align with the thought processes in specialized knowledge areas corresponding to system instructions and learn to respond in various manners according to different instructions.

4 Legal SELF-EXPERTISE Data

We applied the SELF-EXPERTISE methodology to the field of law, where accuracy and reasoning are crucial. We give detailed explanation of the SELF-EXPERTISE augmented instruction dataset.

4.1 Seed Dataset

We used 980 legal seed instruction dataset directly created by legal experts. Details of legal experts we worked with are described in Appendix E. This seed instruction dataset includes 560 legal cases and 916 clauses, and the dataset covers four legal domains: civil law in bar exam, criminal law in bar

Domain	# of Data
Civil law in bar exam	100
Criminal law in bar exam	190
Legislative Information	370
Legal Consultation	320
Total	980

Table 1: The amount of seed data per domain.

# of seed data	980
# of generated data in Step 2	2398
# of generated data in Step 4	19184
avg. instruction length	19.5
avg. input length	31.2
avg output length	144.4

Table 2: Statistics of the generated dataset by applying SELF-EXPERTISE. The length is calculated based on the number of words.

	From Knowledge	From Output
(1) Similarity between {original instruction, input} and {new instruction, input}	**0.66**	0.71
(2) Similarity between {original input} and {new instruction, input}	**0.65**	0.69
(3) Similarity among questions originating from the same source	**0.75**	0.78

Table 3: The similarity in BERT-scores between original data and generated data. We compare new user instructions and inputs generated from knowledge versus directly from output. Higher similarity scores indicate lower diversity in augmentation.

exam, legislative information, and legal consultation. The amount of data per domain is shown in Table 1. We built the data by first extracting outputs containing legal knowledge in the respective domain. Then, user questions were formulated to add user instruction and input, thus constructing a user-oriented legal instruction dataset.

4.2 Data Generation Details

We augmented the dataset through SELF-EXPERTISE based on 980 seed dataset. In Step 1, we used `GPT-3.5-turbo`, and in Steps 2 and 4, we used `GPT-4-preview-1106`[1] for generation. More details of generation models and prompts are presented in Appendix A.2, A.3. After four-step generation, we filtered out data that did not conform to the format. Eventually, we generated dataset of 19k pairs. Basic statistics of the generated data are summarized in Table 2.

[1]https://platform.openai.com/docs/models

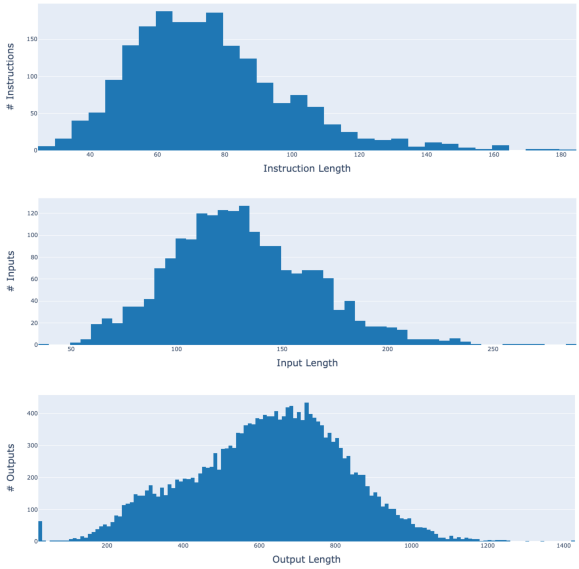

Figure 4: Length distribution of the generated user instructions, inputs, and outputs by SELF-EXPERTISE.

Figure 5: Length distribution of the generated user instructions and outputs by Self-Instruct.

4.3 Diversity

To check the diversity of the generated data, we compared the lengths of the generated user instruction, input, and output. Unlike the Self-instruct generated data, which is biased towards one point, the length distribution of data generated by SELF-EXPERTISE appears to be more even as described in Figure 4 and 5, indicating that it is more diverse. In particular, variously crafted system instructions played an important role in the diversity of output length and form. Depending on each system instruction, the responses could be brief and concise, or they might encompass more detailed and extensive explanations. For instance, we included system instructions that demand core points, as well as those that guide the generation of responses in a step-by-step manner, thereby diversifying the format of the answers.

Data Quality Review Question	Yes %
Does the instruction describe a valid task in the legal field?	90%
Is the input appropriate for the instruction?	90%
Is the output a correct and acceptable response to the instruction and input?	88%
Does the output include correct terms and knowledge?	77%

Table 4: Data quality review results.

Also, in Step 1 of SELF-EXPERTISE, we extracted knowledge from the output. Extracting objective knowledge from outputs first will help model not be limited to a particular situations and create various instructions and inputs. To verify this, we compared the results of generating instructions from the output for a 200 seed dataset with those generated from knowledge. Using BERT-score (Zhang et al., 2019), we measured the similarity between the original instructions, inputs, and the similarity among questions augmented from the same source. As seen in Table 3, instructions generated based on outputs show higher similarity, indicating a reduction in the diversity of the augmented data. Therefore, we decided to proceed to the next step by extracting knowledge from the output in Step 1.

4.4 Quality

Quality of a dataset in the legal field depends on the accuracy of knowledge and logical reasoning, so we conducted human evaluation for the generated data quality measurement. We randomly sampled 100 pairs of user instruction, input, and output. Referring to the data quality review questions (Wang et al., 2022), we asked legal experts to assess whether the sampled data represents valid tasks in the field of law, contains correct legal terms, and includes accurate knowledge. Table 4 shows the results of the human evaluation of the generated data. The evaluation results indicate that generated instructions extensively include law tasks, and the outputs are reasonable and contain accurate legal knowledge. To assist with the understanding, two examples from generated dataset are selected and listed in Appendix A.4.

5 Experimental Setup

5.1 Training Details

To develop LxPERT, we conducted instruction tuning with the LLaMA-2-ko 7B model which is pretrained with Korean language on the LLaMA-2 7B model (Touvron et al., 2023). We trained instruction data augmented with SELF-EXPERTISE using causal language modeling loss. LxPERT underwent three epochs of training on four NVIDIA A100 GPUs with 80GB memory, with the AdamW (Kingma and Ba, 2014) as the optimizer, learning rate of 2e-5, per device train batch size of 1, and a max length of 1024. It took 2 hours to train LxPERT on 19k generated dataset. We enhanced the training speed by utilizing the Accelerate (Gugger et al., 2022) and DeepSpeed [2] libraries. Additionally, we did not use any loss masking and trained the model by calculating the loss from the instruction to the output to generally learn the strategy of producing outputs according to system instructions.

5.2 Baselines

Foundation Models: LLaMA-2 7B (Touvron et al., 2023) and LLaMA-2-ko 7B (L. Junbum, 2023). We chose the 7B-sized sLLM to check the performance of a model of the same size.
Instruction-tuned Models in General Domain: LLaMA-2-chat 7B (L. Junbum, 2023) and LLaMA-2-ko-chat 7B [3]. We selected models trained on existing general domain instruction datasets.
GPT: We included `GPT-3.5-turbo`, the bigger-sized LLM. This research measures and evaluates the performance of GPT model in the Korean legal domain.
Instruction-tuned Models in Legal Domain: Self-Instruct tuned LLaMA-2-ko 7B and seed dataset tuned LLaMA-2-ko 7B. Existing instruction-tuned models have limitations in not being specialized for the Korean legal domain. Therefore, we augmented the dataset for the legal domain by using the Self-Instruct method (Wang et al., 2022) and trained on LLaMA-2-ko 7B. We provide more details of Self-Instruct generation in Appendix D. Also, we add the model trained only seed dataset.

5.3 Evaluation Dataset

In-domain Dataset: In-domain dataset includes new user instruction, input, and output pairs containing legal knowledge from seed dataset. To as-

[2] https://github.com/microsoft/DeepSpeed
[3] https://huggingface.co/heegyu/llama-2-ko-7b-chat

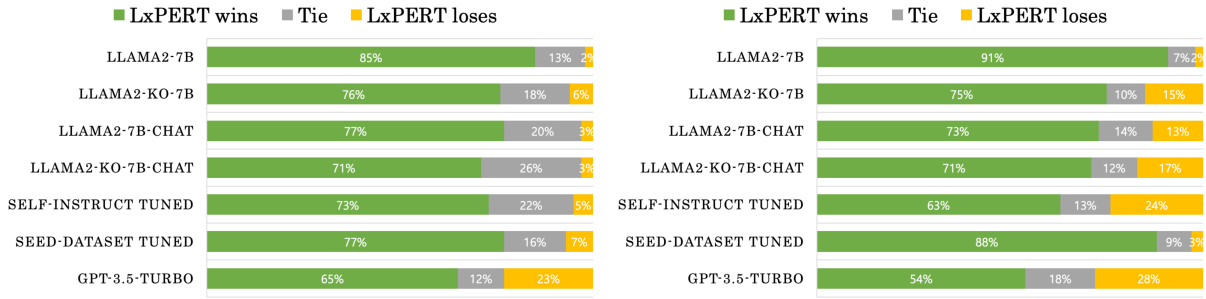

Figure 6: GPT-4 evaluation results on in-domain data (left) and out-of-domain data (right). We pair each response of models with LxPERT response and estimate the win rate.

sess whether the trained model can provide accurate and logical answers reflecting the legal characteristics, we asked legal experts to create a new dataset that is related to same four domains like as seed dataset. Note that, it has been meticulously designed to have no overlap of user instructions, inputs, and outputs with either the seed dataset or the train dataset. Ultimately, test dataset of 200 pairs were compiled.

Out-of-domain Dataset: Out-of-domain dataset means new user instruction, input, and output pairs that contain outside knowledge from seed dataset. To evaluate the model's generalization performance on tasks and knowledge not included in the seed data, we collected 100 QA pairs by crawling the "Easy-to-Find Living Law Information" [4] site. Particularly, we selected questions that require knowledge not presented in our seed data, aiming to evaluate performance on challenging out-of-domain test dataset.

5.4 Evaluation Settings

In order to measure the performance of the model in the legal domain, we conducted GPT automatic evaluation and human evaluation. The model's instruction-following capability was assessed in a zero-shot environment without in-context examples.

GPT-4 Evaluation We conducted pairwise comparison evaluation method used in (Zheng et al., 2024; Wang et al., 2023; Xu et al., 2023). We instructed `GPT-4`[5] to choose the more logical response for the same user instruction and input from two candidate responses. The prompt template used for evaluation is shown in Appendix B.

Human Evaluation Despite GPT's outstanding performance in automatic evaluation, there are limitations in comprehending accuracy and logical structure in specialized domains such as law. Consequently, we asked legal experts to rate the model-generated text on a five-point Likert scale (Likert, 1932) for accuracy and fluency. For accuracy, they assessed whether the included knowledge was correct and appropriate. For fluency, they evaluated whether the task requested in the user instruction was well executed and if the answer was derived from legal reasoning. Detailed criteria is described in Appendix C.3.

6 Results

6.1 Evaluation on In-domain Data

The result of GPT-4 evaluation on in-domain data is shown on the left side of Figure 6. In a pairwise comparison, LxPERT significantly outperforms both models tuned for general domain and those tuned for the legal domain. LxPERT demonstrate superior performance in expanding logical answers through legal reasoning compared to other models. Moreover, LxPERT, which is trained on the characteristics of the legal domain, shows better performance than `GPT-3.5-turbo` which is focused on general domain.

The result of human evaluation is shown in Table 5. Models tuned for general domain instructions score significantly lower, indicating insufficient learning of legal knowledge and thinking, despite possessing some capability in instruction following. Lower performance shows hallucinations in the legal Self-Instruct dataset due to the lack of knowledge. In contrast, LxPERT not only surpasses the performance of other 7B models but also exceeds `GPT-3.5-turbo` in terms of accuracy. LxPERT indirectly acquires legal domain knowledge through an instruction dataset containing accurate knowledge. Furthermore, we observe that

[4] https://www.easylaw.go.kr
[5] https://platform.openai.com/docs/models

Model	In-domain		Out-of-domain	
	Accuracy	Fluency	Accuracy	Fluency
Foundation Models				
LLaMA-2 7B	1.03	1.04	1.04	1.14
LLaMA-2-ko 7B	1.19	1.24	1.06	1.16
Instruction-tuned Models in General Domain				
LLaMA-2-chat 7B	1.63	2.00	1.24	1.76
LLaMA-2-ko-chat 7B	1.61	2.23	1.49	2.37
GPT-3.5-turbo	2.52	3.83	2.60	3.90
Instruction-tuned Models in Legal Domain				
Self-Instruct tuned	1.25	2.06	1.25	2.27
Seed dataset tuned	2.07	3.00	2.88	3.22
LxPERT (Ours)	**3.88**	**4.80**	**2.98**	**4.53**

Table 5: Human evaluation results on in-domain data and out-of-domain data.

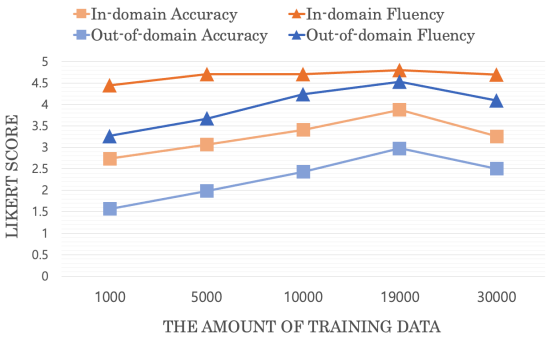

Figure 7: Human evaluation performance of LxPERT tuned with different sizes of training data. Orange lines show the results on in-domain data, while blue lines show the results on out-of-domain data. Triangles mean fluency, and squares mean accuracy.

LxPERT excels in articulating logical structures, including clauses and case law, thus demonstrating an outperforming fluency level.

6.2 Evaluation on Out-of-domain Data

We also measured performance on a challenging out-of-domain test set composed solely of unlearned knowledge. The result of GPT-4 evaluation for out-of-domain (OOD) data is shown on the right side of Figure 6. Among the models trained on the legal domain, the performance of the seed dataset tuned model noticeably drops compared to the result on in-domain data. This could be due to overfitting on a small number of dataset, leading to poor performance in OOD question and answer. While Self-Instruct tuned model and `GPT-3.5-turbo` show a slight improvement in performance in OOD, LxPERET still outperform `GPT-3.5-turbo` with a probability of over 50%.

The human evaluation result for the OOD dataset is shown in Table 5. As before, models including Self-Instruct that underwent instruction tuning show slightly improved performance compared to the baseline. LxPERT, being an out-of-domain set, often based its reasoning on incorrect legal statutes or precedents, resulting in reduced accuracy. Nonetheless, it surpassed `GPT-3.5-turbo`'s performance in this context.

6.3 Quality of Answers Relative to the Amount of Training Data

We measured the relationship between the quantity of augmented instruction dataset and the quality of responses. We increased the dataset sizes to 1k, 5k, 10k, 19k, 30k and observed the performance on in-domain and out-of-domain test set. The results shown in Figure 7. From the graph, we observe that across all models with varying sizes of training data, fluency is consistently higher than accuracy. This trend shows that the model first learns how to response in legal format, regardless of the amount of training data. A general improvement in performance is seen as the amount of training data increases. Notably, there is a decrease in model performance when the data size increases from 19k to 30k. This suggests that excessive augmentation of our limited 980 seed data knowledge leads to overfitting on this specific knowledge, consequently diminishing the model's general linguistic capabilities. Based on these findings, we hypothesize that incorporating additional general domain datasets or expanding knowledge of seed data could further enhance model performance. This hypothesis forms the basis of our proposed future work.

7 Discussion

In our experiments with SELF-EXPERTISE and instruction tuning, we aimed to distill two key attributes: the ability to follow instructions and legal domain knowledge. The model showed proficiency in generating responses based on legal statutes and case law, adhering to the logical framework of legal reasoning, as evidenced by high fluency scores. However, as the model learn the legal domain knowledge indirectly, it was prone to more frequent errors. Prominent errors included hallucinations, where the model fluently provided responses based on inaccurate legal references. This suggests that while SELF-EXPERTISE has addressed knowledge issues compared to Self-Instruct, there are still aspects that remain to be resolved. We anticipate a combination of accurate knowledge acquisition methods with SELF-EXPERTISE to further enhance the precision in legal domain knowledge.

We propose this potential improvement as a subject for future work.

8 Conclusion

In this study, we propose SELF-EXPERTISE for automatically generating instruction dataset in specialized domain areas. Our proposed method extracts knowledge from outputs of a seed dataset and uses this as a basis for generating instructions. This significantly reduces hallucination in instruction dataset creation. To demonstrate the effectiveness of the our augmentation method, we trained LxPERT with an augmented dataset on LLaMA-2 7B and compared it with other baselines. It surpasses the performance of 7B models and `GPT-3.5-turbo` in terms of GPT-4 evaluation and human evaluation. We believe that this methodology can be extended and used for creating instruction datasets not only in the legal domain but also in other specialized knowledge domains. Therefore, we look forward to the utilization of this augmentation pipeline when training an sLLM specialized for knowledge-intensive tasks.

Limitations

This study proposes a methodology for generating instruction datasets in specialized knowledge domains. This methodology has some limitations common to other automatic instruction generation methods and knowledge-based learning methodologies.

Cost Issue The SELF-EXPERTISE process used the `GPT-3.5-turbo` model for Step 1 and the `GPT-4-preview-1106` model's API for Steps 2 and 4. This was necessary for generating accurate responses based on knowledge and creating creative instructions. While it reduces costs compared to human-written data, a significant cost increase can occur when multiple iterations are performed.

Knowledge Expansion Issue We augmented the dataset by creating various instructions based on the knowledge in the seed dataset. This is meaningful as it produces different instructions and inputs from the same knowledge, as discussed in the data analysis section. Moreover, experiments confirm that this augmentation method improves the model's ability to follow precise knowledge and legal narrative structures. However, the major issue is that augmentation with the same seed dataset is only possible with the same knowledge, thus limiting diversity. We believe this can be improved by retrieving a variety of knowledge for data augmentation in future work.

Ethics

The data was generated based on an LLM; therefore, it may contain biases inherent in the backbone LLM. For example, bias may be introduced in the generation of legal questions based on certain knowledge. However, our method is novel in that we did not directly generate inputs from the outputs. Instead, we generated knowledge from the output to create new inputs based on general cases. This process addresses biases and reduces the inclusion of personal information. Additionally, we disclose data augmented with SELF-EXPERTISE. This data is released under the CC-BY-NC 4.0 license [6], which excludes commercial use.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00621,Development of artificial intelli- gence technology that provides dialog-based multi-modal explainability).

References

Akari Asai, Sewon Min, Zexuan Zhong, and Danqi Chen. 2023. Retrieval-based language models and applications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 6: Tutorial Abstracts)*, pages 41–46, Toronto, Canada. Association for Computational Linguistics.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack Rae, Erich Elsen, and Laurent Sifre. 2022. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot

[6] https://creativecommons.org/licenses/by-nc/4.0/

learners. *Advances in neural information processing systems*, 33:1877–1901.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Jonathan H Choi, Kristin E Hickman, Amy Monahan, and Daniel Schwarcz. 2023. Chatgpt goes to law school. *Available at SSRN*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. 2023. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *arXiv preprint arXiv:2306.16092*.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2023. Auggpt: Leveraging chatgpt for text data augmentation.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for nlp. *arXiv preprint arXiv:2105.03075*.

Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. 2022. Accelerate: Training and inference at scale made simple, efficient and adaptable. `https://github.com/huggingface/accelerate`.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Mi-Young Kim and Randy Goebel. 2017. Two-step cascaded textual entailment for legal bar exam question answering. In *Proceedings of the 16th edition of the International Conference on Articial Intelligence and Law*, pages 283–290.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

L. Junbum. 2023. llama-2-ko-7b (revision 4a9993e).

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of psychology*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822.

Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, et al. 2023. Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*.

OpenAI. 2023. Gpt-4 technical report.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, et al. 2020. Kilt: a benchmark for knowledge intensive language tasks. *arXiv preprint arXiv:2009.02252*.

Alexander J Ratner, Henry Ehrenberg, Zeshan Hussain, Jared Dunnmon, and Christopher Ré. 2017. Learning to compose domain-specific transformations for data augmentation. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Mobashir Sadat, Zhengyu Zhou, Lukas Lange, Jun Araki, Arsalan Gundroo, Bingqing Wang, Rakesh R Menon, Md Rizwan Parvez, and Zhe Feng. 2023. Delucionqa: Detecting hallucinations in domain-specific question answering. *arXiv preprint arXiv:2312.05200*.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Yuanhe Tian, Ruyi Gan, Yan Song, Jiaxing Zhang, and Yongdong Zhang. 2023. Chimed-gpt: A chinese medical large language model with full training regime and better alignment to human preferences. *arXiv preprint arXiv:2311.06025*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Andrew Vold and Jack G. Conrad. 2021. Using transformers to improve answer retrieval for legal questions. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, ICAIL '21, page 245–249, New York, NY, USA. Association for Computing Machinery.

Tianlu Wang, Ping Yu, Xiaoqing Ellen Tan, Sean O'Brien, Ramakanth Pasunuru, Jane Dwivedi-Yu, Olga Golovneva, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2023. Shepherd: A critic for language model generation. *arXiv preprint arXiv:2308.04592*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Chenxi Whitehouse, Monojit Choudhury, and Alham Fikri Aji. 2023. Llm-powered data augmentation for enhanced crosslingual performance. *arXiv preprint arXiv:2305.14288*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Wenpeng Yin, Qinyuan Ye, Pengfei Liu, Xiang Ren, and Hinrich Schütze. 2023. LLM-driven instruction following: Progresses and concerns. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, pages 19–25, Singapore. Association for Computational Linguistics.

Fangyi Yu, Lee Quartey, and Frank Schilder. 2023a. Exploring the effectiveness of prompt engineering for legal reasoning tasks. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13582–13596, Toronto, Canada. Association for Computational Linguistics.

Fangyi Yu, Lee Quartey, and Frank Schilder. 2023b. Exploring the effectiveness of prompt engineering for legal reasoning tasks. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13582–13596.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Xujiang Zhao, Jiaying Lu, Chengyuan Deng, Can Zheng, Junxiang Wang, Tanmoy Chowdhury, Li Yun, Hejie Cui, Zhang Xuchao, Tianjiao Zhao, et al. 2023. Domain specialization as the key to make large language models disruptive: A comprehensive survey. *arXiv preprint arXiv:2305.18703*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2020. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008.

Example1 (Civil law in bar exam)

Instruction
Does C have a liability for damages due to unjust enrichment against A?
Input
A entered into a land lease agreement with B regarding B's X land for the construction of a building (lease period from June 1, 2016, to May 31, 2021, 5 years, lease deposit of 700 million won, monthly rent of 20 million won). A completed the construction of Y building on the X land on August 22, 2017, and completed the registration of ownership preservation. A entered into a building lease agreement with C, who intended to operate a warehouse in the Y building (lease period from October 1, 2017, to September 30, 2020, 3 years, lease deposit of 100 million won, monthly rent of 5 million won). Around April 2020, A asked C about extending the lease of the Y building, but C refused to extend it further. Amid the COVID-19 pandemic, with business being non-operational, C, after the expiration of the lease period on September 30, 2020, locked the warehouse door with a key, leaving the goods inside.
Output
It has been established that if a tenant continues to occupy a part of the leased property after the termination of the lease contract as a way of exercising the right of objection to simultaneous performance but does not use or profit from it according to the original purpose of the lease contract, thus gaining no substantial benefit, the tenant does not have the obligation to return any undue gains, even if the landlord has suffered damages (Supreme Court Decision on February 9, 2001, Case No. 2000Da61398). Since C locked the door with a key without taking out the goods, thereby gaining no substantial benefit, C is not obligated to return any undue gains.

Example2 (Legislative Information)

Instruction
I am born while my father and mother were living together. Later, my father married someone else and had two children with them. Recently, as my father was nearing his death, he sought to acknowledge me, but he passed away before I could meet him. According to his will, he recognized me as his child and included me in the inheritance. I am wondering if this is possible.
Output
In order for a child born out of wedlock to be legally recognized as a child of the father, it must be acknowledged either through voluntary acknowledgment or through a lawsuit for acknowledgment (voluntary acknowledgment, paternity suit). As in your case, if the biological father acknowledges his child through a will, the legal effects of the will become effective from the time of the father's death, establishing a legal parent-child relationship between the father and the child (in the case of acknowledgment through a will, the executor of the will must file the acknowledgment report within one month from the date they assume their role). Once a legal parent-child relationship is established, a relationship of inheritance naturally follows. Therefore, as you have been legally acknowledged as a child through your father's acknowledgment, you are entitled to inherit his property along with the other two children. ◇ Lawsuit for Confirmation of a Parent-Child Relationship This is a lawsuit filed to legally confirm the existence of a parent-child relationship between specific individuals. There is no time limit for filing this lawsuit, but if one party to the suit dies, the lawsuit must be filed within two years from the date of knowing about the death.

A Data Generation Details

A.1 Examples of Seed Dataset

We display two examples of our seed dataset, one from 'Civil Law in Bar Exam' and the other one from 'Legislative Information'. For a better understanding, we translated these examples into English.

A.2 Prompts for Generation

SELF-EXPERTISE automatically generates instruction data by giving appropriate prompts to LLMs in each step described in §3.2. In this section, we present three actual prompt templates and eight system instructions. All prompts include 2-

Step1 Prompt Template Example (2-shot)

```
## 1 [output]
{output}
====
Based on the [output], what are the legal facts that can be understood? If there are laws or precedents, please write them down item by item with references. [legal knowledge item list]
[legal knowledge item list]
{knowledge}

## 2 [output]
{output}
====
Based on the [output], what are the legal facts that can be understood? If there are laws or precedents, please write them down item by item with references. [legal knowledge item list]
[legal knowledge item list]
{knowledge}

## 3 [output]
{output}
====
Based on the [output], what are the legal facts that can be understood? If there are laws or precedents, please write them down item by item with references. [legal knowledge item list]
```

Step2 Prompt Template Example (2-shot)

```
## 1
[knowledge]
{knowledge}
You are an exam writer who creates questions based on a knowledge base. You need to appropriately create [question] and [context] based on [knowledge]. Create a new [question] and [context] about new [knowledge]. (It's okay to leave out the context if it's not needed) The more diverse the questions, the better.

[question 1]
{question1}
[context 1]
{context correspond to question1}
====
## 2
[knowledge]
{knowledge}
You are an exam writer who creates questions based on a knowledge base. You need to appropriately create [question] and [context] based on [knowledge]. Create a new [question] and [context] about new [knowledge]. (It's okay to leave out the context if it's not needed) The more diverse the questions, the better.

[question 1]
{question1}
[context1]
<empty>
====
## 3
[knowledge]
{knowledge}

You are an exam writer who creates questions based on a knowledge base. You need to appropriately create [question] and [context] based on [knowledge]. Create a new [question] and [context] about new [knowledge]. (It's okay to leave out the context if it's not needed) The more diverse the questions, the better.
```

Step3 System Instructions

[One Major Instruction]
You are a legal counselor. User will ask you an advice. Your goal is to advise user with a precise legal basis in Korean. When you advise user, you should look up references.

[Eight Sub Instructions]
[1] Provide a detailed and clear advice with confidence so that user can trust you.
[2] Answer in a faithful, kind, friendly manner.
[3] Remind that user does not know any legal information. Think like you are answering to a five-year old.
[4] Answer in a concisely manner, by only including conclusion.
[5] While advising think step-by-step.
[6] Answer in following order: conclusion, basis, advise
[7] Legal reasoning decomposes into four sequential steps: issue, rule, application, conclusion. Follow the steps of legal reasoning to answer.
[8] You do not have much time to answer. Get to the point directly.

Step4 Prompt Template Example (2-shot)

```
## 1 [references]
{knowledge}
[user]
{instruction}
{input}

{system_instruction}
[output]
{output}
===
## 2 [references]
{knowledge}
[user]
{instruction}
{input}

{system_instruction}
[output]
{output}
===
## 3 [references]
{knowledge}
[user]
{user_instruction}
{input}

{system_instruction}
```

shot examples. In case of system instructions, we referred to the styles of system message from Orca (Mukherjee et al., 2023) and modified to suit the legal field.

A.3 Generation Models and Parameters

A.3.1 Generation Models

In this paper, we used `GPT-3.5-turbo` for Step 1 (§3.2.3), and `GPT-4-1106-preview` for Steps 2 (§3.2.2) and 4 (§3.2.4) of the SELF-EXPERTISE application. In Step 1, the task involves extracting and organizing information from given texts rather than creating new content. In this case, we found no significant performance difference between `GPT-3.5-turbo` and `GPT-4-1106-preview`, with both models performing the task effectively. Therefore, we chose the more cost-effective `GPT-3.5-turbo`. However, Steps 2 and 4 relatively require the generation of not only accurate, but also new and creative data from the given texts. In comparison to `GPT-4-1106-preview`, `GPT-3.5-turbo` tended to generate more uniform and similar data rather than a variety. Consequently, `GPT-4-1106-preview` was used for Steps 2 and 4 to facilitate the creation of a more diverse range of data.

A.3.2 Generation Parameters

When conducting the generation, we employed default settings for the parameters. Both frequency penalty and presence penalty were set to 0. Temperature was adjusted to 1, and similarly, top p was configured at 1.

A.4 Examples of Generated Dataset

This section provides two examples of data generated through SELF-EXPERTISE. We translated examples into English to enhance understanding. When looking at the examples, the user instructions include valid questions related to law and also appropriately match with corresponding inputs. In addition, the outputs provide suitable answers to the questions posed in the user instructions, not only responding accurately but also logically explaining based on correct knowledge. This demonstrates the successful implementation of the methodology for creating instruction data based on knowledge.

B GPT-4 Evaluation Details

We conducted the pairwise comparison evaluation between model-generated samples using `GPT-4`. The prompt template for pairwise comparison is shown below. We guide `GPT-4` to choose a more appropriate response based on the user instruction and input. In this case, we assumed that `GPT-4` may not have learned certain legal knowledge, so we include accurate knowledge as a reference. Therefore, `GPT-4` can evaluate based on this knowledge whether the response is accurate and logical.

C Human Evaluation Details

C.1 Human Evaluation Setup

In this section, we present more details for the human evaluation. We conducted two parts of human evaluation: data quality review of the SELF-EXPERTISE augmented dataset, and the evaluation of generated samples by baseline models and LxPERT on the evaluation dataset. To objectively

Example1

User Instruction
Is there any possibility of holding the manufacturer accountable if my car caught fire while it was parked, but I can't prove a defect in the vehicle itself?
Input
My car, which was recently parked, suffered damage due to a fire. The exact cause of the fire has not been identified, and it is unclear whether a specific defect inside the vehicle was the cause of the fire. I have followed the necessary procedures, but the manufacturer is not acknowledging the defect.
Output
In cases where a consumer claims damage based on a manufacturing defect in a vehicle, the consumer must prove that the defect caused the fire. According to the Supreme Court decision on July 28, 2000 (Case No. 98Da35525), it is difficult for the consumer to claim that there is a manufacturing defect in the vehicle if they cannot specify the defect and the possibility of external ignition cannot be ruled out. This applies similarly when asserting the vehicle company's warranty liability for defects. Therefore, in your case, if you cannot clearly identify the cause of the vehicle fire and cannot prove that an internal defect was the direct cause of the fire, it may be very difficult to hold the manufacturer responsible. However, if you cannot completely rule out the possibility of a vehicle defect, further investigation or expert assessment can be pursued. For example, you can seek an expert's opinion on the cause of the fire, or collect claims and evidence from other vehicle owners in similar cases to try to prove a manufacturing defect. If sufficient evidence to prove the manufacturer's responsibility cannot be obtained, depending on the insurance status of the vehicle, claiming insurance for the fire damage could be an alternative.

Example2

User Instruction
Is there an obligation for medical professionals to protect information learned during the diagnosis and treatment process even after the patient's death?
Input
A family's breadwinner passed away while receiving treatment in a hospital. During this process, information about the specific illness they were suffering from was spread among acquaintances not by the family but by a staff member of the medical institution. One of the family members of the deceased patient is looking to take action against the medical institution and that staff member.
Output
Yes, even after a patient's death, healthcare professionals and medical institution employees have a duty not to disclose or publish the patient's information. This duty is established under Article 19, Paragraph 1 of the Medical Law, which prohibits healthcare professionals and medical institution employees from disclosing or publishing secrets learned while performing medical duties. The Supreme Court, in its judgment dated May 11, 2018, stated that the duty of healthcare professionals to keep secrets is intended to protect the trust relationship between the healthcare professional and the patient, and thereby protect and promote the public interest in national health. This judgment includes the secrets of deceased individuals within the scope of information that healthcare professionals are prohibited from disclosing. In the situation you described, the family of a deceased patient can take action against the medical institution or its employees for violating the Medical Law. If a healthcare institution employee leaks patient information, the family can hold that employee criminally responsible and also seek civil damages. The family should first establish the facts and gather evidence, then consult with a legal expert to proceed with the appropriate legal actions.

assess the accuracy of legal knowledge and logical reasoning, we asked two lawyers to evaluate both data quality and model-generated samples.

Prompt for Pairwise Comparison
You are a helpful and precise assistant for checking the quality of the response. Two pieces of response have been provided to the same instruction and input related to law. Which one is more precise and logical refer to the knowledge? Knowledge: {knowledge} Question: {user_instruction} {input} ### Response 1: {model1_output} Response 2: {model2_output} Please choose from the following options. (one option ONLY) Do not generate explanation. A: Response 1 is significantly better. B: Response 2 is significantly better. C: Neither is significantly better.

C.2 Human Evaluation Guideline for Data Quality Review

The guideline for a human evaluation of data quality review is shown below. We provide user instruction, input, knowledge, and output pairs to evaluators. We selected four data quality review questions based on Self-Instruct (Wang et al., 2022) and asked evaluators to answer each question with either 'yes' or 'no'. The result of this human evaluation is presented in Figure 4.

Human Evaluation Guideline for Data Quality Review
You will given a instruction and a corresponding input related to law, and knowledge which can be referred to. You will then given a output to the instruction. Please first read the instruction, input, knowledge and output, and answer yes or no to the questions below: - Does the instruction describe a valid task in the legal field? - Is the input appropriate for the instruction? - Is the output a correct and acceptable response to the instruction and input? - Does the output include correct terms and knowledge?

C.3 Human Evaluation Guideline for Model-generated Samples

The guideline for a human evaluation of model-generated samples is shown below. We gave user instruction, input, knoweldge, and output pairs to evaluators and instructed them to score for accuracy and fluency on a five-point Likert scale (Likert, 1932). The result of this human evaluation is presented in Figure 5.

Human Evaluation Guideline for Model-generated Samples
You will given a question and a corresponding context related to law, and knowledge which can be referred to. You will then given one potential response to the question. Please first read the question, context, and knowledge, and score the generated response on a scale of 1 - 5 on the following criteria: - **Accuracy**: Does the response use correct knowledge in evidence? Does the response answer correctly to the question? - **Fluency**: Is the response appropriate answer to the question? Is the response derived from legal reasoning?

D Legal-based Self-Instruct Details

To compare with the existing instruction data augmentation method, we selected the Self-Instruct method (Wang et al., 2022) and augmented our seed dataset using Self-Instruct. We used `GPT-4-1106-preview` for a fair comparison between Self-Instruct and SELF-EXPERTISE. After six iterations, we created 19k Legal-based Self-Instruct dataset, as same size as SELF-EXPERTISE dataset. Here we show the prompt templates used for generating instruction and instance, respectively.

D.1 Instruction Generation

This stage is for generating instructions based on the tasks in the seed dataset. We used the same prompt template from Self-Instruct.

D.2 Instance Generation

This stage involves creating an instance based on the instruction generated from the previous step. Likewise, we used the same prompt template from Self-Instruct.

Prompts for Legal-based Self-Instruct

Prompt for Instruction Generation
Come up with a series of tasks in Korean: Task 1: {instruction1} (…) Task 8: {instruction8} Task 9:
Prompt for Instance Generation
Come up with example for the following task in Korean. Try to generate multiple examples when possible. If the task doesn't require additional input, you can generate the output directly. Task: {user_instruction} Output: {output} (…) Task: {user_instruction}

E Legal Experts Details

The research project we are currently working on includes a legal team among the participating research teams. This legal team consists of law school professors, students, and lawyers. They are not crowd workers we need to recruit, but co-researchers who perform the same task and receive funding for their research. The dataset creation was led by a law school professor, and the ten law school students helped data writing. Additionally, human evaluation of the dataset quality and model generation results were handled by two lawyers.