Multi-Scale Model Interaction for Refining Financial Consumer Complaints

Anonymous ACL submission

Abstract

Legal writing necessitates a high degree of clarity, formality, and domain-specific precision, attributes frequently absent in documents drafted by non-legal professionals. This paper addresses this disparity by investigating the transformation of informal, colloquial input into persuasive legal arguments. We introduce FinDR, a Chinese dataset of financial dispute records, annotated with official judgments on claim reasonableness. Our proposed method, Multi-Scale Model Interaction (MSMI), leverages a lightweight classifier to evaluate outputs and guide iterative refinement by Large Language Models (LLMs). Experimental results demonstrate that MSMI outperforms single-pass prompting strategies. Additionally, we validate the generalizability of MSMI on several short-text benchmarks, showing improved adversarial robustness. Our findings reveal the potential of multi-model collaboration for enhancing legal document generation and broader text refinement tasks.

Natural Language Inference (NLI) (contradiction -> entailment)	***(original - premise)*** It has long been known that inventories at the macroeconomic level affect the depth and length of business cycles. ***(original - hypothesis)*** It is **unknown** whether business cycles are affected by inventories at the macroeconomic level. **(counterfactual)** It is **known** that business cycles are affected by inventories at the macroeconomic level.
Sentiment Analysis (positive -> negative)	***(original)*** Delicious! The food is always **good and fresh**. Don't go during normal lunch hours though as it gets crazy busy! ***(counterfactual)*** Not good! The food is always **mediocre and not fresh**. Do go during normal lunch hours as it's not busy at all!
Legal Text Refinement (unreasonable -> reasonable)	***(original)*** My insurance agent never informed me about the premium increase. If I had known the premium would go up, I wouldn't have taken the insurance!! ***(refined)*** I was not informed by my insurance agent of the impending premium increase. I would have opted against purchasing the insurance if I had been made aware of the increase. **This omission constitutes a breach of duty on the part of the insurance agent, and I seek appropriate redress for this failure to disclose critical information.**

Figure 1: Comparison of generated counterfactual texts between our purposed task and the other classification tasks. The example of the legal text refinement task is a plaintiff's claim in an insurance dispute, sampled from our dataset.

1 Introduction

People compose a variety of texts in their daily lives, and they often need expert advice for formal documents. In the legal domain, legal writing requires precise presentation of facts, strong argumentation, and the usage of specialized vocabulary (Neumann et al., 2021), which can be challenging for non-professionals. People without a legal background may struggle to express their concerns using legal language, resulting in more conversational texts compared to those written by professionals such as lawyers. This can lead to disadvantages such as applications being more likely to be rejected or claims less likely to be adopted. Moreover, hiring legal experts to assist with such documents can be costly, placing a financial burden on individuals and small businesses. To address this issue, automated tools and methodologies that assist in generating legally sound documents can significantly reduce the need for human intervention, thus lowering costs and improving accessibility to legal expertise.

Recent advancements in Large Language Models (LLMs) have demonstrated strong capabilities in counterfactual generation (Dixit et al., 2022; Chen et al., 2023; Nguyen et al., 2024). These models can produce fluent counterfactual texts that rival or even surpass human-written content. However, most existing work has focused on short texts, such as those found in Natural Language Inference (NLI) or sentiment analysis tasks, as shown in Table 1. These tasks often involve simple modifications like negation or word replacement to change the label of a sentence. In contrast, generating counterfactuals for longer and more complex documents, such as legal texts, presents unique challenges. Legal documents often involve intricate arguments from plaintiffs, defendants, and judges.

Dataset	Samples	Avg. Words	Labels
IMDB (Maas et al., 2011)	50,000	234	positive, negative
Yelp (Zhang et al., 2015)	598,000	133	positive, negative
AGNews (Zhang et al., 2015)	127,600	37	world, sports, business, sci/tech
MNLI (Williams et al., 2018)	431,992	20 / 10	entailment, contradiction, neutral
DBPedia (Auer et al., 2007)	630,000	46	14 labels
FinDR (Proposed)	13,132	598	reasonable, unreasonable

Table 1: Comparing the statistics of FinDR with various popular counterfactual generation datasets. For MNLI, average words is denoted by (premise / hypothesis).

Refining or rewriting such documents requires a deep understanding of the case and modifications at the sentence or paragraph level, rather than simple word-level changes.

To explore this direction, we introduce FinDR, a Chinese legal document dataset composed of **Fin**ancial **D**ispute **R**ecords from real-world cases. FinDR is designed to push the boundaries of counterfactual generation by requiring models to generate more persuasive and formally structured legal claims. Figure 1 shows how counterfactual text generation in FinDR differs from that in typical tasks. In NLI and sentiment analysis, counterfactual rewriting focuses on semantic or sentiment polarity shifts. In contrast, legal text rewriting demands more sophisticated domain-specific knowledge and rigorous logical reasoning. It requires not only adherence to the original factual context but also transforms informal or unreasonable statements into formal claims with legal validity.

In this work, we focus on the task as legal text refinement, where the goal is to transform colloquial writing into polished, formal legal arguments. A central question in refinement is: *"How can we determine whether one version of a text is better than another?"* This is a challenging question, as writing quality can be subjective and context-dependent. To address this, we propose using well-trained classifiers as discriminating models to evaluate the refined output. These models serve as proxies for legal or linguistic judgment, helping to quantify improvements in quality, relevance, and persuasiveness.

To sum up, the contributions of our work are as follows:

1. We propose the task of legal text refinement to explore the transformation of colloquial language into professional legal writing.
2. We introduce FinDR, a challenging Chinese legal document dataset that demands nuanced understanding and sentence-level editing.
3. We explore a novel approach that leverages counterfactual generation with LLMs, using discriminator feedback to guide and evaluate refinement quality.

Label	Count
Unreasonable	10,401
Partially reasonable	1,738
Reasonable	524
Partially unreasonable, partially inapplicable	383
Partially reasonable, unreasonable, and inapplicable	65
Partially reasonable, partially inapplicable	19

Table 2: Dataset Statistics

2 Dataset

We crawled financial dispute records that are public available from Taiwan Financial Ombudsman Institution.[1] Originally, these records were legal documents in pdf format. We extract the content and perform data cleaning to remove texts such as tables, page numbers or links. We collected a total of 13,130 records with dates ranging from February 2012 to March 2023. Each record consists of three primary columns: the plaintiff's claim, the defendant's rebuttal, the judgment made by the officials, and a label.

There are 6 possible labels attached to a record, with the most frequent two labels being "reasonable" and "unreasonable", shown in Table 2. A record is labeled "reasonable" if the officials find the plaintiff's claim to be reasonable and vice versa. The remaining labels include "partially reasonable", "partially unreasonable", and "partially inapplicable", indicating that certain parts of the claim are accepted, rejected and irrelevant, respectively. For the text refinement task, we use a modified version of the dataset where the labels are remapped by the following rule: if the label contains "reasonable" or "partially reasonable", then it will be remapped to "reasonable", otherwise "unreasonable."

[1]The Financial Ombudsman Institution is a government organization funded by Taiwanese government that aims to fairly, reasonably, and effectively resolve financial disputes between consumers and enterprises.

3 Method

We divide the refinement task into two phases: evaluation and refinement. These tasks are assigned to models of different scales. Specifically, a small pre-trained language model (PLM) is employed to evaluate whether an input is reasonable or unreasonable. Based on the evaluation results provided by the PLM, an LLM performs the refinement. Through several rounds of interaction, the objective is to enable the PLM to judge inputs that were initially considered unreasonable as reasonable. We name this approach Multi-Scale Model Interaction (MSMI).

3.1 Discriminating Model for Evaluation

We fine-tune a binary classification model to predict whether a given legal text (typically, a plaintiff's claim) is reasonable or not. The model is trained using our labeled FinDR dataset, where the labels correspond to the judgment outcomes: "reasonable" vs. "unreasonable". We employed the BERT-base model (Devlin et al., 2019) as the PLM and selected the best performing checkpoint on the validation set as the discriminating model M. During refinement, this model provides feedback to the generator, indicating whether a new version of the input text changes the classification result.

3.2 LLM-based Refinement via Counterfactual Generation

To improve an "unreasonable" input t into a "reasonable" version t_c, we employ an LLM as a generator to perform the rewriting. The LLM is prompted with the original text, along with context and instructions. To achieve more effective refinements, we perform multiple rounds of generation and evaluation. After each round, the LLM receives feedback indicating whether the new version was accepted by the classifier. If not, the model is encouraged to revise again, using reinforcement or rejection cues based on the classifier's confidence.

3.3 Iterative Loop and Early Stopping

We adopt an iterative loop similar to the self-refinement setting (Madaan et al., 2023). However, instead of using internal self-evaluation, we use external signals from M. In each iteration:

1. The LLM generates a refined version t_c of t.
2. The classifier evaluates t_c and returns a probability score.
3. If the score crosses a threshold, we stop and accept the result.
4. Otherwise, we provide feedback and loop again.

4 Experiments

To test the effectiveness of MSMI, we sample 100 random samples that are labeled "unreasonable" from our proposed dataset. We use the success rate to measure how effectively our method transforms documents from unreasonable to reasonable refinements. Additionally, we computed the cosine similarity between the original and refined document using BERTScore (Zhang et al., 2020). A high cosine similarity indicates that the refinement preserves the original meaning while making the text more reasonable, whereas a low similarity may suggest substantial rewriting or a shift in semantic content.

To train the discriminating model, we adopt an 80-10-10 split for training, validation, and testing. For the base model, we tested multiple BERT variants, including `google-bert/bert-base-chinese`, `ckiplab/bert-base-chinese` (finetuned by additional traditional Chinese data) and `thunlp/Lawformer` (Xiao et al., 2021), a pre-trained language model designed for processing long Chinese documents. We initially hypothesized that Lawformer might offer advantages due to its extended input limit of 3,072 tokens, compared to the 512-token limit of standard BERT. However, as shown in Table 3, Lawformer slightly underperforms compared to BERT. Moreover, due to its longer input sequences, Lawformer incurs a significantly higher computational cost, averaging roughly six times more processing time than BERT. Therefore, we choose the original BERT as the base model.

As for the LLMs, we used GPT-3.5 (Ouyang et al., 2022), GPT-4o (Hurst et al., 2024), and LLaMA-3 (Grattafiori et al., 2024) for refinement. We also used the approach of simply prompting the LLM without a classifier as the baseline to evaluate the effectiveness of the proposed MSMI method.

Table 4 present the experimental results. First, regardless of LLM is used, the MSMI strategy consistently improves the success rate, demonstrating the generalizability of our method. Second, based on the cosine similarity results, MSMI introduces slightly more content variation, meaning the final

Base Model	Macro-F1
`google-bert/bert-base-chinese` (Devlin et al., 2019)	0.6191
`ckiplab/bert-base-chinese`	0.6063
`thunlp/Lawformer` (Xiao et al., 2021)	0.6073

Table 3: Classifier result for the test set of FinDR.

Strategy	LLM	**Success Rate** (↑)	**Cosine Sim.** (↑)
Prompt	GPT-3.5	8.70%	0.943
	GPT-4o	16.00%	0.974
	LLaMA-3-70B	13.10%	0.957
MSMI (Proposed)	GPT-3.5	25.00%	0.943
	GPT-4o	26.10%	0.962
	LLaMA-3-70B	26.10%	0.946

Table 4: Experimental results in FinDR.

Dataset	**Method**	**Adv Acc.** (↓)	**Cosine Sim.** (↑)
MNLI	GPT-3.5	18.6	**0.96**
	LLaMA-3-70B	27.5	**0.96**
	MSMI (Proposed)	**4.7**	0.80
AG News	GPT-3.5	12.7	0.86
	LLaMA-3-70B	4.0	**0.89**
	MSMI (Proposed)	**3.5**	0.80
Yelp	GPT-3.5	33.0	0.89
	LLaMA-3-70B	**2.8**	0.89
	MSMI (Proposed)	4.4	**0.94**
IMDB	GPT-3.5	10.2	0.90
	LLaMA-3-70B	4.6	0.86
	MSMI (Proposed)	**1.8**	**0.96**
DBPedia	GPT-3.5	36.5	**0.90**
	LLaMA-3-70B	9.9	0.87
	MSMI (Proposed)	**7.1**	0.80

Table 5: Experimental results in short-text refinement tasks.

text tends to undergo greater changes after multiple rounds of modification. Third, we found another advantage of MSMI—it may eliminate the need to select a specific LLM. From the results of the original prompts, there is a noticeable performance gap among different LLMs. However, after applying MSMI, even GPT-3.5 achieves performance comparable to GPT-4o and LLaMA-3.

In sum, the experimental findings validate the effectiveness and robustness of the MSMI method. Not only does it consistently improve performance across different LLMs, but it also helps mitigate disparities between models of varying capabilities. This makes MSMI a practical solution in real-world scenarios where access to the best-performing LLMs may be limited due to cost or availability. These results collectively highlight MSMI as an ideal enhancement strategy for leveraging LLMs in iterative refinement tasks.

5 Further Exploration

As previously mentioned, most earlier tasks focused on short text refinement. In this section, we evaluate the performance of the proposed MSMI model under this specific task. Experiments are conducted using the five benchmarks listed in Table 1. To enable comparison with prior studies, we adopt their methodology by using "Adversarial Accuracy" as the evaluation metric.

Adversarial Accuracy measures a model's robustness against adversarial attacks by evaluating its prediction accuracy on adversarially perturbed inputs. Specifically, it reflects the percentage of adversarial examples on which the model correctly predicts the label. A lower Adversarial Accuracy indicates a more effective attack, as the model is more easily misled by perturbations. Conversely, a higher Adversarial Accuracy suggests that the model is more resilient and better able to withstand adversarial manipulations. Therefore, in our experiments, Adversarial Accuracy is considered a "the lower, the better" metric.

As shown in Table 5, in four out of the five datasets, MSMI outperforms the baseline approach of directly using LLMs. This indicates that, even in short-text scenarios, a single-pass refinement by an LLM may be insufficient. From the perspective of cosine similarity, multiple rounds of refinement tend to result in more substantial content changes—suggesting that the model is not merely inserting "not" or altering a single word, but may be rewriting the text to a greater extent after several iterations. In the context of short texts, this leads to more noticeable differences in cosine similarity. Combining results from both long-text (legal) and short-text experiments, we demonstrate that MSMI is a simple yet effective approach that enhances both refinement quality and adversarial robustness.

6 Conclusion

This paper tackles the challenge of transforming informal legal claims into refined, persuasive documents. We introduce the FinDR dataset, which serves as a benchmark for evaluating legal text refinement. Our proposed MSMI framework demonstrates clear advantages over single-step generation, consistently improving classification success rates while maintaining semantic coherence. Moreover, MSMI proves effective across both long and short-text scenarios, reducing performance gaps among different LLMs and enhancing adversarial robustness. These results suggest that collaborative model architectures, where small models offer targeted feedback to larger generators, hold great promise for advancing automated writing tools.

Limitations

This work focuses on financial consumer disputes in Taiwan, so the current findings remain domain- and language-specific. Text characteristics and reasoning styles in other jurisdictions may differ, which offers opportunities for future cross-domain exploration. The discriminating model used in MSMI is constrained by input length and may not fully capture the structure of long legal documents. Future work could extend this to models with longer context or hierarchical encoding. Because publicly available financial dispute records are limited, the dataset size is relatively small, reflecting a common challenge in real-world legal data and highlighting the need for broader open resources. In addition, expert evaluation is not yet included due to the cost and difficulty of consistent annotation; incorporating professional review would be a valuable next step. Finally, the current label design is binary, which simplifies modeling but leaves room for more fine-grained outcomes in future research.

GenAI Usage Disclosure

We utilized generative AI tools solely for grammar and language refinement. All content was reviewed and edited by the author(s), who take full responsibility for the final manuscript.

References

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The Semantic Web*, pages 722–735, Berlin, Heidelberg. Springer Berlin Heidelberg.

Zeming Chen, Qiyue Gao, Antoine Bosselut, Ashish Sabharwal, and Kyle Richardson. 2023. DISCO: Distilling counterfactuals with large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5514–5528, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Tanay Dixit, Bhargavi Paranjape, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. CORE: A retrieve-then-edit framework for counterfactual data generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2964–2984, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. *Preprint*, arXiv:2310.01798.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Divyansh Kaushik, Eduard Hovy, and Zachary C. Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. *Preprint*, arXiv:1909.12434.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Preprint*, arXiv:2303.17651.

Nishtha Madaan, Inkit Padhi, Naveen Panwar, and Diptikalyan Saha. 2021. Generate your counterfactuals: Towards controlled counterfactual generation for text. *Preprint*, arXiv:2012.04698.

R.K. Neumann, E. Margolis, and K.M. Stanchi. 2021. *Legal Reasoning and Legal Writing*. Aspen Coursebook Series. Wolters Kluwer.

Van Bach Nguyen, Paul Youssef, Jörg Schlötterer, and Christin Seifert. 2024. Llms for generating and evaluating counterfactuals: A comprehensive study. *Preprint*, arXiv:2405.00722.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Bhargavi Paranjape, Matthew Lamm, and Ian Tenney. 2022. Retrieval-guided counterfactual generation for QA. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1670–1686, Dublin, Ireland. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Preprint*, arXiv:2303.11366.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for chinese legal long documents. *Preprint*, arXiv:2105.03887.

Wenda Xu, Guanglei Zhu, Xuandong Zhao, Liangming Pan, Lei Li, and William Wang. 2024. Pride and prejudice: LLM amplifies self-bias in self-refinement. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15474–15492, Bangkok, Thailand. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*.

A Related Work

To obtain better quality of output texts, several works have explored the "self-refining" and "self-correcting" abilities of LLMs, where they revise their responses based on internal feedback to their previous outputs. Madaan et al. (2023) proposed an iterative refinement framework that can obtain better output texts without the need of additional training or fine-tuning. Shinn et al. (2023) takes a step further, leveraging LLMs to convert feedback from the task environment, usually in binary or scalar form, to a verbal text summary, which is then provided as additional context for generation.

However, a few studies have pointed out that these self-correction schemes are not reliable. Huang et al. (2024) showed that without any external or human feedback, self-correcting models are not confident about their response, often revising a correct answer to an incorrect one, thus causing a performance drop. Xu et al. (2024) formalized the evaluation of self-bias of LLMs, and demonstrated that when an evaluation step is included in a self-refine pipeline, the self-bias is amplified, favoring its own generation and deviating from human evaluation. Therefore, we leverage external feedback from trained classifiers or other LLMs to prevent self-bias and potential performance decrease.

Counterfactual data are adversarial inputs designed to flip a model's prediction with subtle changes to the original inputs (Kaushik et al., 2020). In NLP, this involves word perturbations, text additions and deletions, or even the whole structure. While counterfactual generation is often discussed as a method of data augmentation (Madaan et al., 2021; Paranjape et al., 2022), in this paper, we propose leveraging counterfactual generation as a means of text refinement. For example, in binary classification, given an accurate discriminating model M and a text t that is labelled "negative" (denoting "bad text"), the refinement goal is to generate a counterfactual text t_c that causes M to flip its prediction to positive (denoting "good text"), we call t_c a refinement of t.