R-GDA: Reflective Guidance Data Augmentation with Multi-Agent Feedback for Domain-Specific Named Entity Recognition

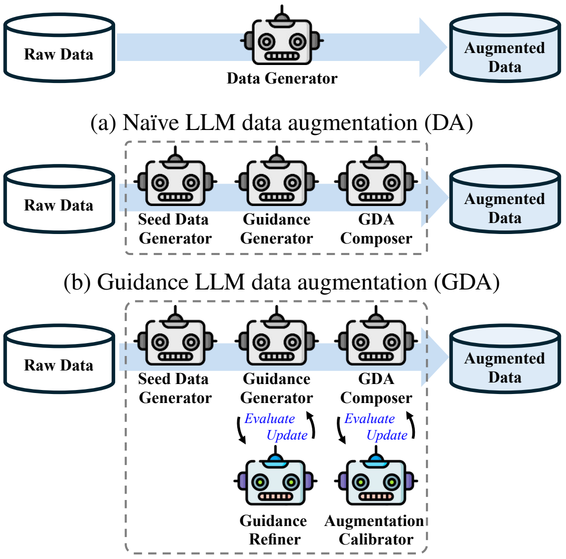
Hyeonseok Kang[1], **Hyuk Namgoong**[1], **Goun pyeon**[1], **Sangkeun Jung**[1*]
[1]Computer Science and Engineering, Chungnam National University, Republic of Korea
{dnfldjaak11,hyuk199,eunbinkim777,hugmanskj}@gmail.com

Abstract

Domain-specific Named Entity Recognition (NER) often requires data augmentation due to the scarcity of annotated corpora. Guidance Data Augmentation (GDA), a method utilizing Large Language Models (LLMs) to decompose sentences into abstract components, can lead to over-abstraction, resulting in undefined entity tags and sentences lacking domain-specific vocabulary. In this work, we propose Reflective GDA (R-GDA), a framework that introduces a multi-agent feedback loop to enhance augmentation quality. R-GDA incorporates two distinct agents: a **Guidance Refiner (GR)**, which assesses the initial abstraction to prevent over-generalization, and an **Augmentation Calibrator (AC)**, which validates the final generated sample for domain-fidelity and tag integrity. On the SciERC and NCBI-disease datasets, R-GDA improves F1-Score, validating its effectiveness. Concurrently, it achieves low BERTScore in most cases, indicating greater sentence diversity. For the FIN dataset, it achieves performance comparable to the GDA baseline. R-GDA consistently prevents errors regarding domain-specific tags, demonstrating that the reflective feedback mechanism enhances data fidelity by mitigating critical generation errors.

1 Introduction

In specialized domains such as healthcare and finance, accurate identification of domain-specific vocabulary and terminology is crucial for reliable information extraction (Durango et al., 2023; Ahmad et al., 2023; Kocaman and Talby, 2022, 2021). Healthcare systems require precise recognition of medical entities including disease names, drug compounds, and clinical procedures from unstructured clinical notes (Liu et al., 2024; Luo et al., 2018), while financial institutions depend on accurate identification of specialized financial terms for compliance and risk assessment (Alvarado et al., 2015). Named Entity Recognition (NER) becomes particularly challenging in these specialized domains where standard vocabularies are insufficient and domain-specific entities exhibit unique linguistic patterns.



(c) Reflective-guidance LLM data augmentation (Ours)

Figure 1: Comparison between data augmentation using LLM, Guidance LLM-based data augmentation and reflective-guidance LLM data augmentation

Recent advances in Large Language Models (LLMs) have prompted investigations of their application for domain-specific NER tasks, yet studies reveal substantial limitations (Hu et al., 2024). LLMs exhibit difficulties in recognizing domain-specific entities due to limited exposure to specialized terminologies during pre-training, resulting in contextual misunderstandings and boundary detection errors (Hu et al., 2024). Developing domain-specific NER systems faces critical challenges including the scarcity of high-quality annotated corpora, which demand expert knowledge and substantial resources (Durango et al., 2023).

[*] Corresponding author

Findings of the Association for Computational Linguistics: EACL 2026, pages 4938–4953
March 24-29, 2026 ©2026 Association for Computational Linguistics

To address these data scarcity issues, researchers have increasingly explored LLM-based data augmentation approaches. LLM-based augmentation methods such as AugGPT have demonstrated effectiveness in few-shot scenarios (Dai et al., 2025; Zhang et al., 2024). Recently, Guidance Data Augmentation (GDA) introduced a promising approach that decomposes sentences into abstracted context, structure, and entity roles to generate varied sentences while maintaining context-entity relationships (Kang et al., 2024). However, existing GDA methods suffer from **over-abstraction** issues where domain-specific characteristics are lost during the abstraction process, resulting in undefined entity tags and generic sentences lacking specialized vocabulary. When domain-specific entities are abstractly generalized, the regeneration process may produce semantically irrelevant tokens or assign incorrect entity boundaries, compromising augmented training data quality.

This study proposes **R**eflective **G**uidance **D**ata **A**ugmentation (R-GDA), an iterative multi-agent framework that addresses these limitations through specialized feedback mechanisms. Building on GDA's theoretical foundation (Kang et al., 2024), our approach introduces two complementary agents that operate in feedback loops to ensure both abstraction quality and augmentation fidelity for domain-specific NER training. As illustrated in Figure 1, R-GDA extends the traditional GDA pipeline by incorporating reflective evaluation and refinement stages that systematically improve augmentation quality. Our framework comprises:

- **Guidance Refiner (GR):** Prevents over-generalization during abstraction by evaluating context quality, keyword detection, structural relationships, entity role clarity, and token-entity mapping accuracy using a systematic scoring framework.
- **Augmentation Calibrator (AC):** Validates domain suitability of generated sentences through comprehensive assessment of domain alignment, semantic consistency, entity role integrity, and tag format validation.

Through iterative refinement, each agent performs multiple evaluation rounds until achieving target thresholds or providing specific improvement guidance for progressive quality enhancement. Unlike the linear data flow in naive DA and standard GDA approaches shown in Figure 1a and Figure 1b, our R-GDA framework (Figure 1c) establishes a feedback loop mechanism that ensures quality control at each augmentation stage. We demonstrate three key contributions: *(i)* maintaining domain-specific semantic consistency while ensuring structural diversity; *(ii)* generating high-quality augmented data that preserves specialized vocabulary and entity patterns; and *(iii)* establishing a quantitative multi-agent quality assurance framework that systematically validates domain suitability through comprehensive evaluation criteria.

Our comprehensive experiments on three domain-specific NER datasets validate the R-GDA framework. Results show R-GDA outperforms the standard GDA baseline, improving the F1-score by up to 0.0396 while simultaneously generating higher-quality data, as evidenced by low BERTScores in most cases. Ablation studies confirm that both the GR and AC agents are crucial to this success. The framework also shows reliability and effectiveness in low-resource settings, demonstrating its practical utility.

2 Related Works

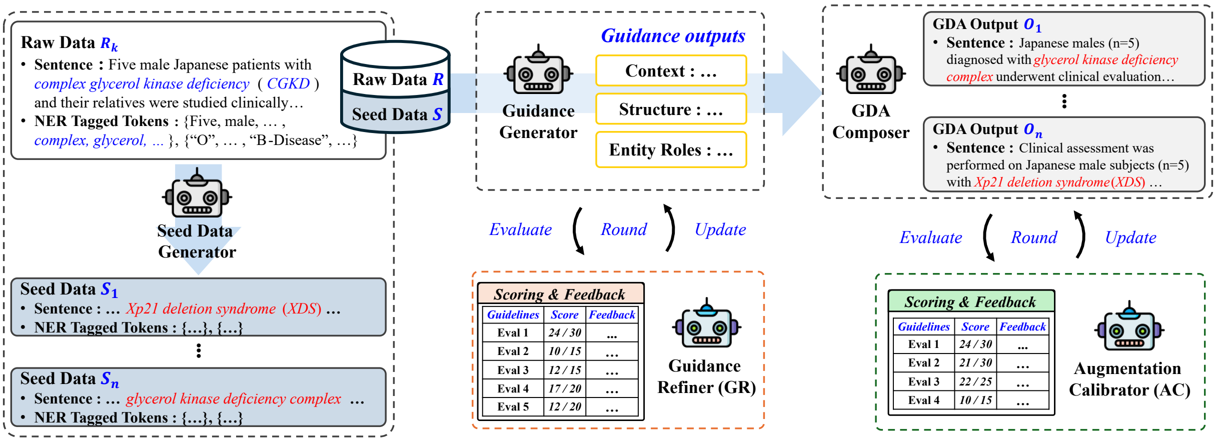
This research is grounded in three key areas: LLM-based data augmentation techniques, data abstraction methods for enhanced reasoning, and LLMs as evaluators of generated content. We review the relevant literature in each area, highlighting the foundations upon which our approach builds.

2.1 LLM-based Data Augmentation

Large language models have been leveraged to generate synthetic training data, reducing reliance on human annotation. Self-Instruct introduced a pipeline where an LLM produces its own instruction-output pairs for self-fine-tuning, achieving near state-of-the-art instruction-following performance (Wang et al., 2023). Building on this, WizardLM employed Evol-Instruct to iteratively rewrite seed instructions into more complex queries, automatically creating diverse datasets (Xu et al., 2024). Models fine-tuned on such LLM-generated data demonstrate performance competitive with human-curated instructions, validating the effectiveness of LLM-based augmentation.

2.2 Data Abstraction for Reasoning

In complex reasoning tasks, encouraging LLMs to abstract problems before solving them improves accuracy. Step-Back prompting guides models to



- **Sentence :** Five male Japanese patients with *complex glycerol kinase deficiency* (*CGKD*) and their relatives were studied clinically…
- **NER Tagged Tokens :** {Five, male, … , *complex, glycerol, …* }, {"O", … , "B-Disease", …}

- **Sentence : …** *Xp21 deletion syndrome* (*XDS*) **…**
- **NER Tagged Tokens : {…}, {…}**

- **Sentence : …** *glycerol kinase deficiency complex* **…**
- **NER Tagged Tokens : {…}, {…}**

- **Sentence :** Japanese males (n=5) diagnosed with *glycerol kinase deficiency complex* underwent clinical evaluation…

- **Sentence :** Clinical assessment was performed on Japanese male subjects (n=5) with *Xp21 deletion syndrome*(*XDS*) …

Guidelines	Score	Feedback
Eval 1	24 / 30	…
Eval 2	10 / 15	…
Eval 3	12 / 15	…
Eval 4	17 / 20	…
Eval 5	12 / 20	…

Guidelines	Score	Feedback
Eval 1	24 / 30	…
Eval 2	21 / 30	…
Eval 3	22 / 25	…
Eval 4	10 / 15	…

Figure 2: Illustration of prompt flows for guidance data augmentation in NER tasks using feedback agents.

first derive high-level concepts from input questions, then use these abstractions to inform step-by-step reasoning (Zheng et al., 2023). By focusing on sentence-level abstraction and defining the reasoning scope, the model reduces errors from overly detailed chains-of-thought. Empirical results show that PaLM-2 with Step-Back prompting achieves substantial accuracy gains on physics and timeline reasoning tasks, highlighting abstraction's role in enabling interpretable and effective problem-solving.

2.3 LLMs as Evaluators

Recent work explores using LLMs to evaluate or critique generated content in place of human judges. *Reinforcement Learning from AI Feedback (RLAIF)* uses LLMs as reward providers, achieving performance on par with RLHF (Lee et al., 2023). Similarly, Constitutional AI employs principle-guided LLMs to automatically judge and refine responses (Bai et al., 2022). Beyond training-time feedback, LLMs serve as metrics for text generation quality—GPTScore evaluates text by prompting models to rate along desired criteria (Fu et al., 2024), while G-EVAL uses GPT-4 with chain-of-thought rubrics, showing higher correlation with human judgments than traditional metrics (Liu et al., 2023). These approaches illustrate the growing trend of using LLMs to audit and improve the outputs of other LLMs.

Using insights from these earlier studies, our approach addresses the limitations of over-abstraction in existing GDA methods by incorporating multi-agent feedback mechanisms that systematically refine and validate both the abstraction quality and the augmented samples.

3 Reflective Guidance Data Augmentation with Multi-Agent Feedback

In this study, we propose an R-GDA framework that addresses the limitations of over-abstraction of existing GDA approaches through a systematic multi-agent feedback mechanism. As illustrated in Figure 2, the workflow begins with the Seed Data Generator module, which samples labeled seed instances from the training data and produces the initial augmented candidates. These candidates are subsequently refined through three core components: (1) *Guidance Refiner*, which evaluates and refines abstraction quality to prevent over-generalization while preserving domain characteristics, (2) *Augmentation Calibrator*, which validates the domain suitability and entity integrity of generated augmented samples, and (3) *Iterative Refinement Loop*, which coordinates both agents through multiple evaluation rounds to ensure progressive quality enhancement.

3.1 Guidance Refiner (GR)

The Guidance Refiner (GR) prevents over-abstraction during the process by evaluating guidance quality across multiple dimensions. In practice, this issue manifests as excessive generalization of domain-specific terminology and the generation of entity types outside the defined schema—for instance, producing hallucinated labels such as ARTICLE or STATUS instead of the target NER tags (e.g., PER, LOC, ORG, and MISC). Such errors directly degrade the utility of augmented data for downstream NER models. In our framework, guidance corresponds to "summarizing" and "structuring" the input sentence with respect to the required NER entity types and domain-specific information.

To align with this goal of summarization and structuring, we draw on text summarization evaluation methodologies (Isonuma et al., 2021; Lindemann et al., 2023) and comprehensive quality assessment frameworks (Güney et al., 2024), the GR ensures that abstracted information maintains domain specificity while providing clear, schema-adherent augmentation directions.

The evaluation is based on a 100-point scoring rubric with weights empirically derived to prioritize key aspects of abstraction quality for NER. The detailed evaluation criteria for the GR are presented in Figure 8 in the appendix.

3.2 Augmentation Calibrator (AC)

The Augmentation Calibrator (AC) validates domain suitability and entity integrity of generated samples. Based on semantic consistency evaluation principles from summarization research and text quality assessment frameworks (Güney et al., 2024), the AC evaluates whether the augmented sentence preserves the roles that entities play in the sentence, the overall sentence structure, the contextual information needed for domain-specific interpretation, the use of specialized vocabulary, adherence to the target NER tagging requirements, and sufficient lexical and structural diversity.

Similar to the GR, the validation uses a 100-point scoring system with empirically calibrated weights to prioritize domain content preservation and semantic integrity. The complete rubric for the AC is shown in Figure 9 in the appendix. The full prompts for each generation step (Seed, Guidance, and GDA Composer) are also available in Appendix A.

3.3 Iterative Refinement Loop

The iterative refinement mechanism, as formalized in Algorithm 1, coordinates both GR and AC agents through systematic evaluation rounds, ensuring progressive quality enhancement until the target thresholds are achieved or maximum iteration limits are reached. This process prevents infinite loops while maintaining quality standards through structured feedback incorporation.

As detailed in Algorithm 1, the refinement process begins with initial input evaluation using the specified agent type (GR for guidance refinement or AC for augmentation calibration). Each evaluation round generates both a numerical score (0–100) and detailed textual feedback identifying specific improvement areas. The loop terminates when the score reaches a target threshold or a maximum number of rounds is completed.

Algorithm 1: Iterative Refinement Loop

```
Require : $input$: Initial guidance or augmented data
Require : $type$: GR (Guidance Refiner) or AC (Augmentation Calibrator)
Ensure  : Refined output after up to 3 rounds or if score $\geq 90$

$output \leftarrow input$;
$score \leftarrow 0$ ;    // Initialize score to enter the loop
$round \leftarrow 0$;
repeat
    $round \leftarrow round + 1$;
    $(score, feedback) \leftarrow$ EVALUATE$(output, type)$;
    if $score < 90$ and $round < 3$ then
        $output \leftarrow$ REFINE$(output, feedback)$;
    end
until $score \geq 90$ or $round = 3$;
return $output$;
```

The termination conditions balance quality assurance with computational efficiency: the process concludes when either the target quality threshold (90 points) is achieved or the maximum iteration limit (3 rounds) is reached. In our pilot experiments, we evaluated performance across various thresholds (85, 90, and 95). Lowering the threshold to 85 reduced refinement rounds but led to a higher tag error rate and degraded downstream NER performance. Conversely, a threshold of 95 or higher frequently resulted in samples failing to reach the target within the three-round limit, incurring unnecessary API costs without yielding proportional performance gains. Based on these observations, we established 90 as the optimal threshold to balance augmentation quality, computational cost, and tagging integrity.

The iterative mechanism enables cumulative quality improvements through multiple refinement cycles, where each round builds upon previous feedback to address residual quality issues. This progressive enhancement approach is particularly crucial for domain-specific NER tasks, where maintaining the balance between abstraction generality and domain specificity requires careful calibration across multiple evaluation perspectives.

4 Experimental Setup

In this section, we outline the experimental setup designed to evaluate the effectiveness of the R-GDA framework in domain-specific NER data augmentation.

4.1 Datasets

The dataset configuration for training and evaluating the proposed R-GDA framework was designed following the experimental protocol established in the original GDA study (Kang et al., 2024), ensuring direct comparability with baseline methods. In selecting datasets for our experiments, we focused on data from specialized domains characterized by the specificity and expertise required in their entities, necessitating specialized knowledge for effective data augmentation.

We utilized three datasets representing distinct specialized domains: SciERC (Luan et al., 2018), NCBI-disease (Doğan et al., 2014), and FIN (Salinas Alvarado et al., 2015). These domains are characterized by the expertise and specialized knowledge required for their entities, which is also necessary for effective Data Augmentation (DA). To ensure experimental consistency with the baseline GDA approach (Kang et al., 2024), we maintained identical dataset configurations and augmentation ratios across all experiments.

For each dataset, we randomly extracted 200 seed data instances from the original training set to serve as the foundation for data augmentation. Following the established protocol (Kang et al., 2024), we performed data augmentation to generate 600 additional samples, resulting in a total training dataset of 800 instances per domain. This 1:3 ratio of seed data to augmented data was consistently applied across all three datasets (SciERC, NCBI-disease, and FIN) to ensure fair comparison and validate the generalizability of our approach across different specialized domains. The augmented dataset composition enables comprehensive evaluation of how R-GDA performs under data-scarce conditions typical of specialized domain applications.

4.2 Models

Data Augmentation LLMs Within our proposed R-GDA framework, we employed three state-of-the-art large language models to ensure comprehensive evaluation across different architectural approaches: GPT-4o (OpenAI, 2023), Gemini 2.5-Flash (DeepMind, 2025), and Claude-3.7 Sonnet (Anthropic, 2025). All models were configured with a temperature parameter set to the default value of 1.0 to maintain consistent generation behavior across experiments. To prevent the influence of differences in language understanding and ensure fair comparison, the same LLM version was used for both abstraction and augmentation output generation during the guidance data augmentation process.

The selection of these three models represents different paradigms in LLM architecture and training methodologies: GPT-4o represents the latest advancement in OpenAI's GPT series with enhanced multimodal capabilities, Gemini 2.5-Flash provides Google's optimized approach for efficient processing, and Claude-3.7 Sonnet offers Anthropic's constitutional AI approach. This diverse model selection enables comprehensive validation of R-GDA's effectiveness across different LLM architectures and training paradigms.

Evaluation Model for NER Task For the evaluation phase, we utilized pre-trained language models, specifically BERT (Devlin et al., 2019) (bert-base-uncased), following the evaluation protocol established in the original GDA study (Kang et al., 2024). A comparative study was conducted to analyze the training effects of different DA methods across the three specialized domain datasets. The evaluation model was trained using a combination of 200 seed data instances and 600 augmented samples generated through the respective DA methods, maintaining consistent training conditions across all experimental configurations.

Evaluation Metrics Performance evaluation employs a dual-metric approach to comprehensively assess both the effectiveness and diversity of augmented data: F1-score serves as the primary metric for evaluating NER task performance. This metric measures the model's ability to correctly identify and classify domain-specific entities when trained on augmented data. A higher F1-score indicates that the augmented data effectively captures the essential patterns and characteristics needed for accurate entity recognition, thus demonstrating the quality of the augmentation method in preserving task-relevant information. BERTScore evaluates the augmented sentences themselves by measuring their semantic similarity to the original seed data. Unlike the F1-score which assesses downstream task performance, BERTScore directly quantifies the diversity of generated augmentations. Lower BERTScore values indicate reduced similarity be-

Model	Method	Config	F1-Score (↑)			BERTScore (↓)		
			SciERC	NCBI-disease	FIN	SciERC	NCBI-disease	FIN
-	EDA	-	0.5434	0.8062	0.7953	0.986	0.987	0.981
GPT-4o	Naïve DA	-	0.5308	0.7697	0.8520	0.921	0.944	0.985
	GDA	-	0.5159	0.7875	**0.8544**	0.831	0.861	0.813
	R-GDA	GR+AC	0.5380	0.7889	0.8483	0.713	0.756	**0.759**
		w/o AC	0.5212	0.7825	0.8131	0.790	0.801	0.768
		w/o GR	0.5283	0.7736	0.8382	0.789	0.818	0.792
Gemini-2.5 Flash	Naïve DA	-	0.5263	0.7727	0.8412	0.897	0.901	0.921
	GDA	-	0.5067	0.7921	0.8498	0.812	0.853	0.821
	R-GDA	GR+AC	**0.5463**	**0.8182**	0.8474	**0.719**	**0.750**	0.783
		w/o AC	0.5294	0.8035	0.8387	0.821	0.815	0.832
		w/o GR	0.5418	0.8112	0.8417	0.835	0.852	0.861
Claude-3.7 Sonnet	Naïve DA	-	0.5217	0.7615	0.8313	0.910	0.924	0.945
	GDA	-	0.5103	0.7736	0.8335	0.831	0.860	0.803
	R-GDA	GR+AC	0.5321	0.7628	0.8320	0.817	0.756	0.814
		w/o AC	0.5187	0.7612	0.8311	0.852	0.823	0.825
		w/o GR	0.5160	0.7487	0.8209	0.876	0.855	0.827

Table 1: Performance comparison of different methods on three NER datasets with F1 and BERT scores. All experiments were conducted three times with different random seeds, and we report the average scores. The highest performance for each model is highlighted in underline, and the highest performance across all models is indicated in **bold**.

tween augmented and seed sentences, signifying higher linguistic diversity in the augmented dataset. This diversity is crucial for training robust NER models that can generalize beyond the limited patterns present in the original seed data.

This dual-metric approach enables comprehensive assessment of the quality-diversity trade-off, where optimal augmentation methods should achieve high F1-scores (maintaining task effectiveness) while obtaining low BERTScores (ensuring diverse training samples).

5 Experimental Results

In this section, we introduce the experiments designed to validate the effectiveness of the R-GDA framework in enhancing domain-specific NER data augmentation quality. The objectives of our experiments are as follows:

- Demonstrate the superiority of multi-agent feedback by comparing R-GDA performance against baseline GDA and Naïve DA methods across three specialized domain datasets.
- Evaluate the individual contributions of GR and AC agents through ablation studies to verify the effectiveness of each component.
- Analyze the augmentation quality improvements by measuring both task performance (F1-score) and diversity metrics (BERTScore) across different LLM architectures.
- Investigate the relationship between seed data size and augmentation effectiveness to validate the reliability of the proposed framework under varying data scarcity conditions.

5.1 Performance Comparison Across Methods and Models

To evaluate the R-GDA framework's capacity for quality data augmentation, it is essential to compare our proposed method against established baselines, ranging from the rule-based EDA(Wei and Zou, 2019) to LLM architectures, across multiple domain-specific datasets. As shown in Table 1, the R-GDA method with both GR and AC agents demonstrates strong performance in generating diverse yet domain-accurate augmented samples. We further verify the robustness of the gains in Table 1 using paired bootstrap significance tests; details are provided in Appendix B. The complete R-GDA framework achieved lower BERTScores in most configurations, indicating the highest augmentation diversity while maintaining domain specificity.

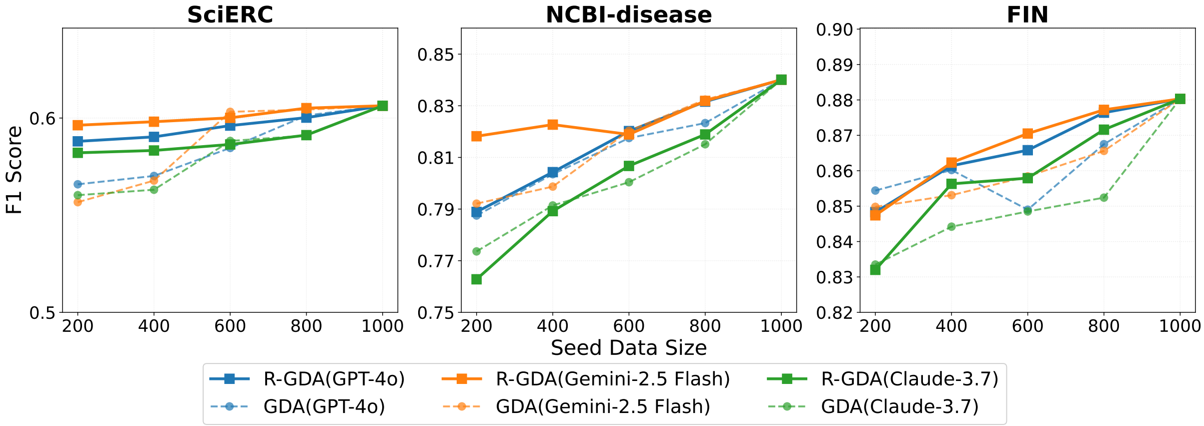


Figure 3: Performance Comparison: GDA vs R-GDA across different seed data sizes.

Notably, R-GDA with Gemini-2.5 Flash achieved the highest F1-scores of 0.5463 on SciERC and 0.8182 on NCBI-disease datasets, outperforming baseline methods.

Across all tested models (GPT-4o, Gemini-2.5 Flash, and Claude-3.7 Sonnet), R-GDA showed improved or competitive performance compared to the baseline GDA method in most experiments. The ablation studies reveal that both GR and AC agents contribute meaningfully to the overall performance: configurations without AC (w/o AC) and without GR (w/o GR) consistently showed degraded performance compared to the complete R-GDA framework, confirming the necessity of both feedback mechanisms for optimal augmentation quality.

5.2 Augmentation Quality Analysis with Varying Data Ratios

The proposed R-GDA method was assessed by measuring performance improvements under different seed data size configurations compared to the baseline GDA approach. As shown in Figure 3, R-GDA demonstrates noticeable performance improvements when augmented data constitutes a higher proportion of the training set. Across all three datasets (SciERC, NCBI-disease, and FIN), the R-GDA method exhibited more gradual and stable performance improvements as seed data size increased, indicating that the augmented data generated by R-GDA maintains higher quality compared to GDA.

This result suggests that our proposed method generates higher-quality augmented data compared to baseline approaches. The consistent performance curves across different seed data ratios demonstrate that R-GDA produces high-quality augmented samples that effectively complement the original training data. When augmented data comprises a larger portion of the training set, R-GDA's advantage becomes more pronounced, validating the framework's effectiveness in generating domain-appropriate samples that preserve specialized vocabulary and entity patterns while ensuring structural diversity necessary for robust model training.

5.3 Qualitative Analysis of Augmentation Quality

To further understand the performance improvements demonstrated by R-GDA, we conducted a qualitative analysis of the augmented samples generated by different methods. Table 2 illustrates a representative example from the SciERC dataset, revealing critical differences in tagging quality between GDA and R-GDA approaches. Additional examples from the NCBI-disease and FIN datasets, which show similar patterns of tag hallucination by GDA and its prevention by R-GDA, are available for review in Appendix C. The examples in Appendix C show that R-GDA, compared to GDA, reduces tag hallucinations while providing greater semantic diversity in the expanded sentences.

The baseline GDA method often generates entity tags not defined in the dataset schema, a phenomenon we term "tag hallucination." As shown in Table 2, GDA introduces erroneous "Effect" tags in the SciERC dataset. This issue arises when the abstraction process loses domain-specific constraints.

In contrast, R-GDA's Augmentation Calibrator (AC) validates tag format compliance, ensuring adherence to the original tag set. The prevention of such tagging errors contributes to R-GDA's improved performance, as the noise from erroneous

Seed data (SciERC)		
Sentence: With a simple two-point calibration, these effects can efficiently be suppressed.		
NER Tags: ['O', 'O', 'O', 'B-Method', 'I-Method', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O']		
Method	**Augmented Data**	
	Sentences	**NER Tags**
GDA	A controlled multi-stage annealing process is optimized to effectively reduce *residual stress within advanced alloys.*	['B-Method', 'I-Method', 'I-Method', 'I-Method', 'O', 'O', 'O', 'O', 'O', '**B-Effect**', '**I-Effect**', '**I-Effect**']
	Quantum tunneling current suppression circuitry is implemented to mitigate *electron leakage in nanoscale devices.*	['B-Method', 'I-Method', 'I-Method', 'I-Method', 'O', 'O', 'O', 'O', '**B-Effect**', '**I-Effect**', '**I-Effect**']
R-GDA (GR+AC)	With an optimized temperature compensation, sensor drift can be alleviated.	['O', 'O', 'O', 'B-Method', 'I-Method', 'O', 'O', 'O', 'O']
	A rapid power spectral density analysis allows for efficient reduction of measurement noise.	['O', 'O', 'B-Method', 'I-Method', 'I-Method', 'O', 'O', 'O', 'O', 'O', 'O', 'O']

Table 2: Comparison of augmented samples generated by GDA and R-GDA methods for a seed sentence from the SciERC dataset using Gemini-2.5 Flash. **Bold** tags indicate erroneously generated entity types that do not exist in the original SciERC annotation schema.

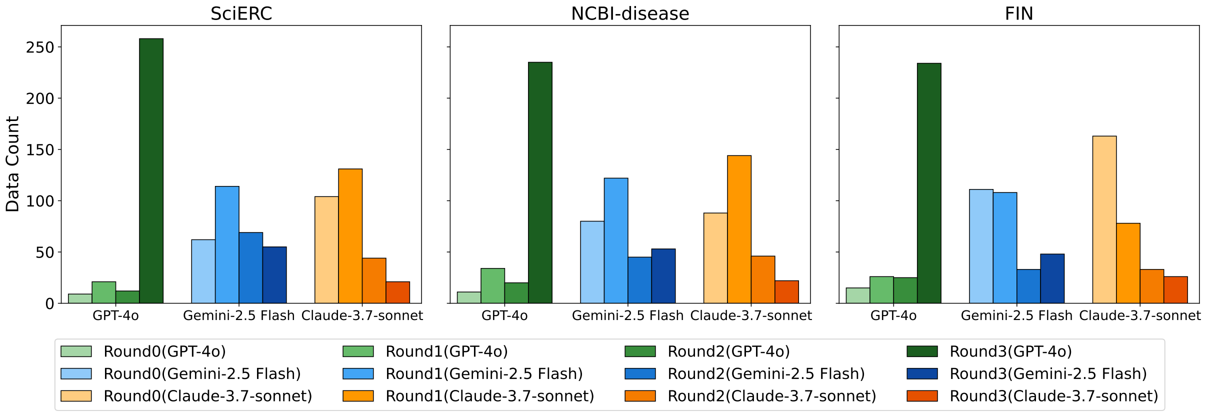


Figure 4: Distribution of refinement iteration counts for each model across the SciERC, NCBI-disease, and FIN datasets. The bars show the number of data points accepted after 0, 1, 2, or 3 rounds of refinement.

tags in GDA-generated samples can otherwise degrade model performance, particularly at higher augmentation ratios.

5.4 Analysis of Refinement Cost and Efficiency

Figure 4 illustrates the trade-off between refinement efficiency, cost, and performance. **Gemini-2.5 Flash** provides the optimal balance; its extremely low per-round cost (Appendix D) allows the R-GDA framework to iteratively refine its outputs to achieve the highest F1-scores, making it ideal for cost-sensitive applications. In contrast, **GPT-4o** exhibits higher initial quality, requiring the fewest refinement iterations. This makes it a strong choice when minimizing API calls is the primary goal, despite its higher cost. **Claude-3.7-Sonnet** proved less cost-effective, requiring more iterations at a higher cost for comparable performance.

6 Conclusion

R-GDA utilizes two reflective agents, a Guidance Refiner (GR) and an Augmentation Calibrator (AC), within an iterative loop to improve the quality of LLM-generated data. The feedback process is designed to curb over-generalization, eliminate spurious tags, and increase syntactic variety without sacrificing domain-specific terminology, which contributes to improved performance on downstream NER tasks. Our experiments show the performance gains from R-GDA are more pronounced when augmented data constitutes a larger portion of the training set. The framework's consistent performance across diverse LLM architectures (GPT-4o, Gemini-2.5 Flash, and Claude-3.7 Sonnet) demonstrates its model-agnostic applicability for various domain-specific NER scenarios. While our approach improves augmentation quality, its limitations include the computational overhead from iterative refinement and the use of fixed scoring thresholds. Beyond NER, the framework illustrates the potential of using score-driven critics with generative models for reliable data augmentation. Future work will explore adaptive scoring policies, extensions to other tasks such as relation extraction and summarization, and dynamic threshold adjustments to optimize the balance between computational efficiency and augmentation quality.

Limitations

In this study, the evaluation was confined to three specialized datasets, excluding standard benchmarks like BC5CDR, and was conducted using a general-purpose encoder without verifying performance on domain-specific models like SciBERT. Furthermore, the study lacks a comparison against other contemporary feedback methodologies, such as self-verification or hybrid approaches. From a practical standpoint, the framework introduces significant computational overhead and relies on fixed heuristics that may require tuning. Finally, the performance is fundamentally dependent on the LLM-as-judge, for which a robustness analysis was not conducted. Future work will aim to address these limitations by expanding the evaluation scope, optimizing the framework's efficiency, and rigorously analyzing the robustness of its components.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(No. RS-2025-0055621731482092640101), Institute of Information and Communications Technology Planning and Evaluation(IITP) grant funded by the Korea government(MSIT) (RS-2019-II190004, Development of semi-supervised learning language intelligence technology and Korean tutoring service for foreigners), Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2022-00155857, Artificial Intelligence Convergence Innovation Human Resources Development (Chungnam National University)) and research fund of Chungnam National University.

References

Pir Noman Ahmad, Adnan Muhammad Shah, and KangYoon Lee. 2023. A review on electronic health record text-mining for biomedical name entity recognition in healthcare domain. In *Healthcare*, volume 11, page 1268. MDPI.

Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the australasian language technology association workshop 2015*, pages 84–90.

Anthropic. 2025. Claude 3.7 sonnet and claude code. Accessed: 2025-08-01.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, and 1 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Fang Zeng, Wei Liu, and 1 others. 2025. Auggpt: Leveraging chatgpt for text data augmentation. *IEEE Transactions on Big Data*.

Google DeepMind. 2025. Gemini 2.5: Our most intelligent ai model. https://blog.google/technology/google-deepmind/gemini-model-thinking-updates-march-2025/. Blog post. Accessed 18 May 2025.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. Ncbi disease corpus: A resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10.

María C Durango, Ever A Torres-Silva, and Andrés Orozco-Duque. 2023. Named entity recognition in electronic health records: a methodological review. *Healthcare informatics research*, 29(4):286–300.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2024. GPTScore: Evaluate as you desire. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6556–6576, Mexico City, Mexico. Association for Computational Linguistics.

Emin Güney, Cüneyt Bayılmış, Serap Çakar, Erdeniz Erol, and Özhan Atmaca. 2024. Autonomous control of shore robotic charging systems based on computer vision. *Expert Systems with Applications*, 238:122116.

Yan Hu, Qingyu Chen, Jingcheng Du, Xueqing Peng, Vipina Kuttichi Keloth, Xu Zuo, Yujia Zhou, Zehan Li, Xiaoqian Jiang, Zhiyong Lu, and 1 others. 2024. Improving large language models for clinical named entity recognition via prompt engineering. *Journal of the American Medical Informatics Association*, 31(9):1812–1820.

Masaru Isonuma, Junichiro Mori, Danushka Bollegala, and Ichiro Sakata. 2021. Unsupervised abstractive opinion summarization by generating sentences with tree-structured topic guidance. *Transactions of the Association for Computational Linguistics*, 9:945–961.

Hyeonseok Kang, Hyein Seo, Jeesu Jung, Sangkeun Jung, Du-Seong Chang, and Riwoo Chung. 2024. Guidance-based prompt data augmentation in specialized domains for named entity recognition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 665–672, Bangkok, Thailand. Association for Computational Linguistics.

Veysel Kocaman and David Talby. 2021. Biomedical named entity recognition at scale. In *Pattern Recognition. ICPR International Workshops and Challenges*, pages 635–646, Cham. Springer International Publishing.

Veysel Kocaman and David Talby. 2022. Accurate clinical and biomedical named entity recognition at scale. *Software Impacts*, 13:100373.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and 1 others. 2023. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*.

Matthias Lindemann, Alexander Koller, and Ivan Titov. 2023. Compositional generalization without trees using multiset tagging and latent permutations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14488–14506, Toronto, Canada. Association for Computational Linguistics.

Shengyu Liu, Anran Wang, Xiaolei Xiu, Ming Zhong, and Sizhu Wu. 2024. Evaluating medical entity recognition in health care: entity model quantitative study. *JMIR Medical Informatics*, 12(1):e59782.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232, Brussels, Belgium. Association for Computational Linguistics.

Ling Luo, Zhihao Yang, Pei Yang, Yin Zhang, Lei Wang, Hongfei Lin, and Jian Wang. 2018. An attention-based bilstm-crf approach to document-level chemical named entity recognition. *Bioinformatics*, 34(8):1381–1388.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90, Parramatta, Australia.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2024. Wizardlm: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*.

Meishan Zhang, Gongyao Jiang, Shuang Liu, Jing Chen, and Min Zhang. 2024. Llm-assisted data augmentation for chinese dialogue-level dependency parsing. *Computational Linguistics*, 50(3):867–891.

Huaixiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou. 2023. Take a step back: Evoking reasoning via abstraction in large language models. *arXiv preprint arXiv:2310.06117*.

A Methodology Details and Prompts

This section details the system prompts that structure the entire data augmentation pipeline. The figures below sequentially present the prompts for each key stage: initial seed augmentation (Figure 5), semantic abstraction via the Guidance Generator (Figure 6), sentence creation by the GDA Composer (Figure 7), and the evaluation rubrics for the Guidance Refiner (Figure 8) and Augmentation Calibrator (Figure 9).

B Paired Bootstrap Significance Tests

To assess whether the performance gains of R-GDA over GDA are statistically reliable, we conduct paired bootstrap resampling on the fixed test set for each experimental setting in Table 1. Specifically, we resample test sentences with replacement (keeping the sample size equal to the original test set) and compute the distribution of ΔF1 = F1(R-GDA) - F1(GDA) over 10,000 bootstrap replicates. We report two-sided p-values as $p = 2 \cdot \min\{\Pr(\Delta \leq 0), \Pr(\Delta \geq 0)\}$ and visualize the resulting ΔF1 distributions in Figure 10.

C Additional Qualitative Comparison of Augmented Samples

This section provides additional qualitative examples of augmented data generated by the GDA and R-GDA frameworks. The following tables complement the SciERC analysis in Section 5.3 by showing results for the NCBI-disease and FIN datasets. In both cases, the baseline GDA method generates tags that are not defined in the respective dataset schemas (highlighted in red), whereas R-GDA consistently adheres to the correct tag set. As presented in the following tables, these cases illustrate both reduced hallucinations and more diverse yet schema-consistent expressions.

D Computational Cost Analysis

This section provides a detailed breakdown of the input costs associated with a single data augmentation round for both the baseline GDA and our proposed R-GDA frameworks.

Table 5 outlines the costs for three different large language models, highlighting the significant cost-effectiveness of Gemini-2.5 Flash. It is important to note that these calculations are based solely on the token count of the static **system prompts** used in each step. The actual cost per round will be higher, as it will also include the tokens from the user-provided seed sentence and any additional context. Furthermore, this analysis only covers input costs; total operational costs will also include the expense of generating the output tokens, which varies based on the length and quantity of the augmented data.

Seed Generation Prompt

```
### NER_TAGGED_SENTENCE ###
{NER tagged sentence}

You are a data augmentation expert for Named Entity Recognition (NER) tasks.

### ROLES ###
- Given a tokenized sentence with entity annotations, generate 3 augmented versions optimized for NER tasks.
- The number of outputs {num_augmented} is provided.
- Preserve entity roles and structure, regardless of tagging scheme (e.g., BIO, BILOU, span-based, etc.).

### REQUIREMENTS ###
- Ensure grammaticality and natural fluency.
- Use vocabulary consistent with the source domain.
- Vary syntax and word choice while keeping entity semantics intact.
- Match the input format: token list and corresponding entity tags.
```

Figure 5: The system prompt used to generate initial augmented sentences from a single seed sentence.

Guidance Generator Prompt

```
### AUGMENTED DATA ###
{augmented_data}

===

You are a semantic abstraction expert for Named Entity Recognition (NER) data.

### ROLES ###
- Given a sentence and its NER-tagged tokens, extract the broader category and semantic structure for data augmentation.
- Focus on identifying the domain category and semantic relationships that can support diverse sentence generation.
- Do not rely on external background knowledge— base all reasoning solely on the sentence itself.

### INTERPRETATION GUIDELINES ###
- **Context**:
  Identify the broader domain category or field where this sentence belongs (e.g., medical diagnosis, financial analysis, scientific research, business communication).
  Provide a concise, abstract description of the sentence's domain context.

- **Structure**:
  Describe the semantic relationships between key elements (subjects, actions, objects, modifiers) in abstract terms.
  Focus on the logical structure that can be applied to generate similar sentences in the same domain.

- **Entity Roles**:
  For each named entity, describe the token, its NER tag (including the tag label and tagging scheme), and its semantic role in the sentence.
  Format as: "token (tag_label, tagging_scheme): semantic_role. token (tag_label, tagging_scheme): semantic_role."
```

Figure 6: The system prompt for the Guidance Generator, which creates a semantic abstraction from a given sentence.

GDA Composer Prompt

```
### GUIDANCE_OUTPUT###
{Guidance Outputs}

===
You are a data augmentation expert for Named Entity Recognition (NER) tasks.

### GOAL ###
Generate {num_augmented} unique, well-formed sentences that reflect the abstract meaning and semantic structure of the original sentence, using the provided interpretation.

### AUGMENTATION INSTRUCTIONS ###
- Each generated sentence must align with the **context** and follow the described **structure**.
- Replace original **entity tokens** with new tokens of the **same NER tag type**, while keeping the entity's **semantic role** consistent.
- Do not reuse original tokens.
- Use grammatically correct and fluent language.
- Ensure the sentence remains relevant to the context and domain.
```

Figure 7: The system prompt for the GDA Composer, which generates new sentences based on the refined guidance.

GR Evaluator Prompt

```
### AUGMENTED DATA ###
{augmented_data}

### GUIDANCE OUTPUT ###
{guidance_output}
===
You are a strict evaluator for semantic abstractions of NER data. Evaluate based on abstraction quality, not literal accuracy
Provide concise, actionable feedback for each criterion. Be extremely rigorous in scoring - deduct points for ANY issues, even minor ones.

### EVALUATION CRITERIA ###

1. **Context Abstraction (30 points)**
  - Reward: ONLY if the context provides detailed domain analysis but remains compact (1-2 lines shorter than current verbose outputs).
  - Penalize: ANY overly verbose descriptions, excessive technical details, or lack of domain-specific characteristics. Deduct heavily for unnecessarily long explanations.

2. **Keyword Detection (15 points)**
  - Reward: ONLY if domain-relevant vocabulary identification and semantic relationships between key terms provide clear augmentation direction.
  - Penalize: ANY missing domain-specific terminology, failure to capture logical relationships, or insufficient detail for augmentation. Deduct for incomplete analysis.

3. **Structure Relationship (15 points)**
  - Reward: ONLY if grammatical roles and logical structure provide explicit guidance for both Raw Input and Augmented Input cases.
  - Penalize: ANY ambiguous structural relationships, unclear augmentation patterns, or failure to address both input types. Deduct for vague descriptions that don't support both cases.

4. **Entity Explanation (20 points)**
  - Reward: ONLY if entity descriptions analyze tokens based on their NER tags and tagging schemes, explaining roles and semantic characteristics by tag type.
  - Penalize: ANY generic entity descriptions, missing tag-based analysis, or lack of semantic role explanation. Deduct heavily for incomplete tag-based token analysis.

5. **Correct Relationship (20 points)**
  - Reward: ONLY if token-entity mapping is completely accurate with proper NER tag correspondence and domain-appropriate boundaries.
  - Penalize: ANY incorrect associations, mismatched tags, boundary errors, or domain misunderstandings. Deduct for any tagging issues.

### FEEDBACK REQUIREMENTS ###
- Provide SPECIFIC, ACTIONABLE guidance for improving each criterion
- Focus on how to make the abstraction more useful for data augmentation
- Give concrete examples of what should be added, changed, or clarified
- Ensure feedback directly supports the goal of generating diverse, domain-appropriate sentences
- For Context: Guide toward detailed but compact domain analysis
- For Entity Explanation: Emphasize tag-based token analysis and semantic role identification
- For Structure: Ensure analysis supports both Raw Input and Augmented Input cases
```

Figure 8: The system prompt for the GR Evaluator, outlining the five criteria for assessing the quality of the semantic abstraction.

AC Evaluator Prompt

```
### GUIDANCE_OUTPUT###
{Guidance Outputs}

### PREVIOUS GDA OUTPUT ###
{GDA_output}
===
You are a strict evaluator for NER data augmentation quality. Evaluate each result based on the following criteria.
Provide concise, actionable feedback for each criterion. Be extremely rigorous in scoring: deduct points for ANY issues, even minor ones.

### EVALUATION CRITERIA ###

1. **Domain Alignment (30 points)**
   - Reward: ONLY if the augmented sentences capture domain-specific contextual meaning and characteristics while maintaining diverse structural and lexical compositions.
   - Penalize: ANY sentences that deviate from domain context, use inappropriate terminology, or fail to reflect domain-specific patterns. Deduct heavily for generic expressions that lack domain authenticity.

2. **Semantic Consistency (30 points)**
   - Reward: ONLY if core elements (subject, action, object, modifiers) and logical relationships (cause, condition, contrast, etc.) are preserved or naturally transformed while maintaining domain-specific semantic coherence.
   - Penalize: ANY loss of core elements, unnatural relationships, or semantic drift that affects domain understanding. Deduct for logical inconsistencies or meaning distortion.

3. **Entity Role Integrity (25 points, NER Task Essential)**
   - Reward: ONLY if entity types and roles are consistently maintained across all variations, preserving original semantic roles and relationships within the domain context.
   - Penalize: ANY change in entity roles, relationships, or semantic functions that alters the original meaning or domain-specific entity behavior. Deduct heavily for entity role confusion.

4. **Tag Format Validation (15 points, NER Task Essential)**
   - Reward: ONLY if BIO/BILOU tagging format is completely correct with clear entity boundaries and consistent tag application across all variations.
   - Penalize: ANY tagging format errors, boundary issues, inconsistent tag usage, or domain-inappropriate entity segmentation.Deduct for any tagging irregularities.

### FEEDBACK REQUIREMENTS ###
- Provide SPECIFIC, ACTIONABLE guidance for improving each criterion
- Focus on how to enhance domain-specific contextual meaning and characteristics
- Give concrete examples of what should be added, changed, or clarified
- Ensure feedback directly supports the goal of generating diverse, domain-appropriate sentences with accurate NER tagging
- For Domain Alignment: Guide toward capturing domain-specific patterns and terminology
- For Semantic Consistency: Emphasize maintaining logical relationships within domain context
- For Entity Role Integrity: Focus on preserving domain-specific entity behaviors and relationships
- For Tag Format Validation: Ensure proper NER tagging that reflects domain entity boundaries
```

Figure 9: The system prompt for the AC Evaluator, detailing the four criteria for validating the quality and domain-fidelity of the final augmented sentences.

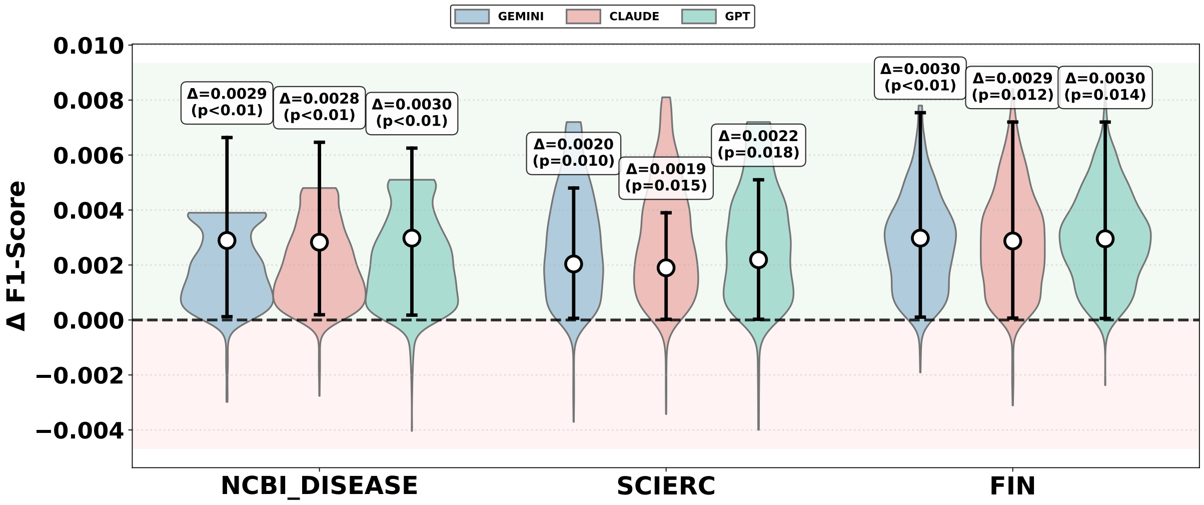


Figure 10: Paired bootstrap significance tests for the ΔF1 improvements of R-GDA over GDA across datasets (NCBI_DISEASE, SCIERC, FIN) and LLM backbones (Gemini, Claude, GPT). Violin plots show the bootstrap distribution of ΔF1 over 10,000 sentence-level resamples of the test set (sample size preserved); white circles indicate the mean ΔF1 and black bars indicate the 95% confidence interval. The dashed horizontal line denotes ΔF1=0. Reported p-values are from a two-sided paired bootstrap test.

Seed data (NCBI-disease)

Sentence: Further investigation of the HEXA gene intron 9 donor splice site mutation frequently found in non- Jewish Tay- Sachs disease patients from the British Isles.

NER Tags: [’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’B-Disease’, ’I-Disease’, ’I-Disease’, ’I-Disease’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’]

GDA	A study commenced on the genetic variant found on the CFTR gene related to the F508del mutation , which is widely observed in the Caucasian population presenting with Cystic fibrosis.	[’O’, ’O’, **’B-GENE’**, ’O’, ’O’, ’O’, **’B-MUTATION’**, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, **’B-DEMOGRAPHIC’**, ’O’, ’O’, ’B-DISEASE’]
	Research was initiated on the G2019S mutation within the LRRK2 gene , commonly reported among the Ashkenazi Jewish individuals suffering from Parkinson’s disease.	[’O’, ’O’, ’O’, ’O’, ’O’, **’B-MUTATION’**, ’O’, ’O’, ’O’, **’B-GENE’**, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, **’B-DEMOGRAPHIC’**, ’O’, ’O’, ’B-DISEASE’]
R-GDA (GR+AC)	Further inquiry into the SCN1A gene missense variant often implicates it in Dravet Syndrome among patients from North America.	[’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’B-Disease’, ’I-Disease’, ’O’, ’O’, ’O’, ’O’, ’O’]
	A deeper look at the cystic fibrosis transmembrane conductance regulator mutation regularly is a marker for Cystic Fibrosis in individuals from Northern Europe.	[’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’B-Disease’, ’I-Disease’, ’O’, ’O’, ’O’, ’O’, ’O’]

Table 3: Qualitative comparison for a seed sentence from the **NCBI-disease** dataset. Erroneously generated tags not present in the original schema are highlighted in **red**.

Seed data (FIN)

Sentence: (i) The Borrower shall make the first repayment on the First Repayment Date.

NER Tags: [’O’, ’O’, ’O’, ’O’, ’B-PER’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’]

GDA	The Obligor must finalize the principal payment no later than the Eleventh of May.	[’B-PER’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, **’B-DATE’**, **’I-DATE’**, **’I-DATE’**]
	Mr. Smith is obligated to make the repayment on the Tenth of June.	[’B-PER’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, **’B-DATE’**, **’I-DATE’**, **’I-DATE’**]
R-GDA (GR+AC)	(i) The Payer is required to complete the disbursement on the Closing Date.	[’O’, ’O’, ’O’, ’B-PER’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’]
	(i) The Debtor must make the installment on the Maturity Date.	[’O’, ’O’, ’O’, ’B-PER’, ’O’, ’O’, ’O’, ’O’, ’O’, ’O’]

Table 4: Qualitative comparison for a seed sentence from the **FIN** dataset. Erroneously generated tags not present in the original schema are highlighted in **red**.

Step	**Gemini-2.5 Flash**	**GPT-4o**	**Claude-3.7-Sonnet**
Seed Data Generation	\$0.000045	\$0.000373	\$0.000447
Guidance Generation	\$0.000071	\$0.000590	\$0.000708
+ GR	\$0.000200	\$0.001665	\$0.001998
GDA Composer	\$0.000048	\$0.000400	\$0.000480
+ AC	\$0.000214	\$0.001785	\$0.002142
Total Cost (GDA)	**\$0.000164**	\$0.001363 (+\$0.001199)	\$0.001635 (+\$0.001471)
Total Cost (R-GDA)	**\$0.000578**	\$0.004813 (+\$0.004235)	\$0.005775 (+\$0.005197)

Table 5: System Prompt Input Cost Comparison per Round for GDA vs R-GDA, Highlighting the Cost-Effectiveness of Gemini-2.5 Flash.