Knowledge-Infused Prompting: Assessing and Advancing Clinical Text Data Generation with Large Language Models

Ran Xu, Hejie Cui, Yue Yu, Xuan Kan, Wenqi Shi, Yuchen Zhuang, May D. Wang, Wei Jin, Joyce C. Ho, Carl Yang

♡ Emory University ♦ Georgia Institute of Technology
{ran.xu, hejie.cui, xuan.kan, wei.jin, joyce.c.ho, j.carlyang}@emory.edu
{yueyu, wshi83, yczhuang}@gatech.edu

Abstract
Clinical natural language processing faces challenges like complex medical terminology and clinical contexts. Recently, large language models (LLMs) have shown promise in this domain. Yet, their direct deployment can lead to privacy issues and are constrained by resources. To address this challenge, we delve into synthetic clinical text generation with LLMs for clinical NLP tasks. We propose an innovative, resource-efficient approach, CLINGEN, which infuses knowledge into the process. Our model involves clinical knowledge extraction and context-informed LLM prompting. Both clinical topics and writing styles are drawn from external domain-specific knowledge graphs and LLMs to guide data generation. Our extensive empirical study across 8 clinical NLP tasks and 18 datasets reveals that CLINGen consistently enhances performance across various tasks by 7.7%-8.7% on average, effectively aligning the distribution of real datasets and enriching the diversity of generated training instances. Our code is available at https://github.com/ritaranx/ClinGen.

1 Introduction
Clinical Natural Language Processing (NLP) emerges as a distinct subfield including the extraction, analysis, and interpretation of unstructured clinical text (Wornow et al., 2023). Despite its significance, unique challenges exist for methodology development in clinical NLP. For example, clinical texts are often dense with abbreviations and specialized medical terminologies can be perplexing to standard NLP models (Lee et al., 2023). Fortunately, recent advances in Large Language Models (LLMs) (Brown et al., 2020; Chung et al., 2022; Ouyang et al., 2022; OpenAI, 2023b,a) provide a promising way to resolve these issues, as they contain billions of parameters and have been pretrained on massive corpora, thus inherently capture a significant amount of clinical knowledge (Agrawal et al., 2022; Singhal et al., 2023). These progresses inspire the need for designing specialized approaches for adapting LLMs to clinical settings, which both address the terminology complexities and improve models through clinical data finetuning (Tu et al., 2023; Liu et al., 2023).

Despite the strong capacity of general LLMs, directly applying them to infer over clinical text data is often undesired in practice. Firstly, these LLMs often have billions of parameters that translate to significant computational resources even for inference, leading to increased infrastructure costs and long inference time. Furthermore, the sensitive patient information in the clinical text naturally raises privacy and regulatory compliance concerns (Meskó and Topol, 2023). To combat these challenges, generating synthetic training data using LLMs serves as a promising solution, as it leverages the capability of LLMs in a resource-efficient and privacy-centric way. When trained with synthetic data mimicking real-world clinical data, models can achieve high performance while obeying data protection regulations.

Synthetic data generation with LLMs is a popular research area in NLP (Meng et al., 2022; Ye et al., 2022a,b; Wang et al., 2023), with a focus on general-domain data. However, adapting LLMs trained on general texts for generating high-quality clinical data poses distinct challenges. To assess the quality of data generated by existing methods, we carry out an evaluation centered on distribution and diversity, detailed in Section 3, which indicate a noteworthy data distribution shift. We further examine the clinically-related entities' quantities and frequencies in synthetic data, where a notable decline is observed when contrasting synthetic data with ground truth data. While some research has delved into clinical data generation with language models, many of these efforts are tailored to specific tasks. Examples include medical dialogues (Chintagunta et al., 2021), clinical
notes (Giorgi et al., 2023), and electronic health records (Ive et al., 2020; Wang and Sun, 2022). These studies often directly adopt language models for text generation, and sometimes on excessive training data. Till now, a unified principle to better adapt LLMs for generating synthetic text for facilitating clinical downstream applications is still missing.

Motivated by the above analysis, we propose CLINGEN, a clinical knowledge-infused framework for high-quality clinical text generation in few-shot scenarios. Our ultimate goal is to bridge the gap between synthetic and real data while enhancing topic diversity. Towards this end, we propose to utilize clinical knowledge extraction to contextualize the prompts. This includes generating clinical topics on entity and relation information from both KGs and LLMs and deriving writing style suggestions from LLMs. By doing this, CLINGEN integrates both non-parametric insights from external clinical knowledge graphs with the intrinsic parametric knowledge encoded in LLMs and enjoys higher diversity via dynamically composing different topics and writing styles together during the data generation process. It is worth noting that, CLINGEN only relies on minimal additional human efforts, and can be readily applied to a wide array of core tasks in clinical NLP.

Our contributions can be summarized as follows:

- We propose CLINGEN, a generic clinical knowledge-infused framework for clinical text data generation in few-shot settings. It can be readily applied to a wide range of tasks in clinical NLP.
- We present an analysis of the pitfall of existing data generation approaches for clinical text data, and propose a simple yet effective strategy to extract clinical knowledge and customize the prompts toward target clinical NLP tasks. This includes generating clinical topics from both KGs and LLMs and deriving writing style suggestions from LLMs.
- We conduct an exhaustive evaluation of synthetic clinical data generation across 8 clinical NLP tasks and 18 datasets. Empirical findings demonstrate that CLINGEN not only aligns more closely with the distribution of the original data but also amplifies the diversity of the generated training samples. The empirical performance gains are consistent across various tasks with different LLMs and classifiers (8.7% for PubMedBERT\textsubscript{Base} and 7.7% for PubMedBERT\textsubscript{Large}).

2 Related Work

Generating additional training data enables a more precise analysis of medical text, and has gained more attention in the past years. Earlier research has employed data augmentation techniques to generate similar samples to existing instances with word substitution (Kang et al., 2021), back translation (Xie et al., 2020), pretrained transformers (Kumar et al., 2020; Zhou et al., 2022). But they often yield rigid transformations and the quality of the augmented text cannot be always guaranteed.

The emergence of LLMs has presented new possibilities for synthetic data generation (Meng et al., 2022, 2023; Ye et al., 2022a; Li et al., 2023). However, these methods often use generic and simple prompts that may not fully capture domain-specific knowledge, thus potentially limiting the quality of the generated data. Liu et al. (2022a); Chung et al. (2023); Yu et al. (2023) employ interactive learning to generate instances, at the cost of additional human efforts. Several recent studies explore LLM-based synthetic data generation for clinical NLP. Tang et al. (2023) rely on a much larger training set to generate candidate entities, which disregards the practical low-resource setting (Perez et al., 2021). Moreover, these studies often concentrate on specific target tasks, thus lacking generality for diverse clinical NLP scenarios.

On the other hand, several works aimed at optimizing prompts using LLMs (Zhou et al., 2023; Wang et al., 2024) or knowledge graphs (Liu et al., 2022b; Chen et al., 2022b), yet they mainly focus on refining prompts to obtain the answer for the given input, and the prompt template often remains unchanged. Instead, we focus on the different task of generating training instances. By composing different topics and styles together, we can generate diverse templates for prompting LLMs to improve the quality of the synthetic data.

3 Preliminary Study

This section first presents the foundational setup of synthetic data generation. Then, we provide an in-depth investigation into the pitfalls of existing synthetic data generation methods.

3.1 Problem Setup

In this paper, we study synthetic data generation under the few-shot setting. The input consists of a training set \( D = \{(x_i, y_i)\}_{i=1}^{K} \), where \((x_i, y_i)\) represents an input text and its corresponding label
Denote the task-specific prompts for class label name $j$ as $p_j$, we take a closer look at the synthetic text data generated by two representative approaches: ZeroGen (Ye et al., 2022a), which directly instructs LLMs for data generation, denoted as $D = \{(\tilde{x}_i, \tilde{y}_i)\}_i^{N}$, where $N$ is the number of generated samples ($N \gg K$). We use $\rho(...)$ to denote the generation process from the LLM. For each downstream task, we fine-tune a classifier $C_{\theta}$ (a moderate-size pretrained language model) parameterized by $\theta$ on the synthetic dataset $D$ for evaluating its quality.$^1$

### 3.2 Limitations of Existing Methods

Denote the task-specific prompts for class label $j$ as $p_j$, we take a closer look at the synthetic text data generated by two representative approaches: ZeroGen (Ye et al., 2022a), which directly instructs LLMs for data generation, denoted as $D_{\text{Zero}} \sim \rho_{j \sim \mathcal{Y}}(\cdot; p_j)$, and DemoGen (Yoo et al., 2021; Meng et al., 2023), which augments the prompt with few-shot demonstrations $D$ as $D_{\text{Demo}} \sim \rho_{j \sim \mathcal{Y}}(\cdot; [p_j, D])$. The prompt format of ZeroGen and DemoGen are in Appendix E.3. We observe that these methods often introduce distribution shifts and exhibit limited diversity, which can lead to suboptimal downstream performance.

**Distribution Shift.** An inherent issue when adapting LLMs to specific domains for text generation is the distribution shift, given that LLMs are primarily trained on vast amounts of web text in general domains. To quantify the data distribution shift, we employ Central Moment Discrepancy (CMD) (Zellinger et al., 2017) to measure the gap between synthetic and real data across six clinical NLP datasets — a high CMD value indicates a large gap between two distributions.\(^2\) Figure 1(a) illustrates that both ZeroGen and DemoGen exhibit elevated CMD scores. Despite the inclusion of few-shot demonstrations in DemoGen, this limitation remains evident, indicating a notable disparity between the ground-truth and synthetic data.

**Limited Diversity.** Clinical datasets in real-world scenarios often include rich domain knowledge that can be challenging to replicate in synthetic data. We evaluate synthetic dataset diversity by using both entity quantity and their normalized frequencies. The results are illustrated in Figures 1(b) and 1(c). Our analysis reveals that datasets generated by ZeroGen and DemoGen exhibit a limited number of clinical entities, having a substantial discrepancy with the ground truth. Furthermore, it is highlighted that only a minority of potential entities and relations are frequently referenced across instances, while the majority are generated infrequently.

To explicitly illustrate the limitations, we present a case study in Figure 9, Appendix B. The comparison reveals that samples generated by ZeroGen and DemoGen lack sufficient details present in the ground truth data. Besides, the generated samples adhere to a more uniform style, while the ground truth encompasses various situations and writing styles, including urgent and informal inquiries.

### 4 Knowledge Infused Data Generation

Section 3 highlights the necessity of domain-tailored knowledge for clinical synthetic data generation. In pursuit of this, we present CLINGEN, a knowledge-informed framework for clinical data generation. The overview of CLINGEN is shown in Figure 2. This two-step methodology harnesses the emergent capabilities of LLMs and external knowledge from KGs to facilitate the synthesis of clinical data, even with few-shot examples only.

#### 4.1 Clinical knowledge extraction

Contrary to previous studies (Ye et al., 2022a,b; Meng et al., 2023) which employ generic queries $p_j$
to prompt LLMs for text generation, CLINGEN emphasizes refining clinically informed prompts. This approach aims to extract rich clinically relevant knowledge from parametric (e.g., LLMs) or non-parametric sources (e.g., knowledge graphs) and tailor it to clinical NLP tasks. To realize this, our modeling contains two dimensions including clinical topics $\mathcal{T}$ and writing styles $\mathcal{W}$, which are integrated into the original prompts to infuse domain-specific knowledge. The Clinical topic refers to a clinical entity (e.g., disease) or relation (e.g., the relationship between diseases and medications), which is usually a phrase, while the writing style is a phrase that depicts the tone, and overall presentation of the text. By composing different topics and writing styles together, CLINGEN provide a diverse suite of prompts, resulting in a wider spectrum of text produced from the LLM $M$. For details of prompt formats across various tasks, please see Appendix E.

### 4.1.1 Clinical Topics Generation

We provide two choices to generate clinical topics $\mathcal{T}$—one is to sample related entities or relations from external KG, and the other is to query relevant knowledge from LLM.

**Topics $\mathcal{T}_{\text{KG}}$ sampled from Non-Parametric KGs.** Healthcare KGs offer a rich collection of medical concepts and their complex relationships, which organizes medical knowledge in a structured way (Li et al., 2022). In our study, we employ the integrative biomedical knowledge hub (iBKHK) as the KG (Su et al., 2023) $G$ to generate topics $\mathcal{T}_{\text{KG}} \sim \text{query}(G)$ due to its broad coverage over clinical entities. To illustrate, for the Disease Recognition task (NCBI, Dogan et al. (2014)), we extract all disease nodes $e$ from the iBKHK to bolster the medical information as $\mathcal{T}_{\text{KG}}^{\text{NCBI}} \sim \text{query}(G_{\text{disease}})$.

$G_{\text{disease}} = \{ e \in G \mid \text{type}(e) = \text{disease} \}$. As another example, we retrieve links between chemicals $c$ and diseases $d$ for the chemical and disease relation extraction (CDR, Wei et al. (2016)) as $\mathcal{T}_{\text{CDR}}^{\text{KG}} \sim \text{query}(G_{\text{relation}\_cd})$, $G_{\text{relation}\_cd} = \{ \{ c, r, d \} \in G \mid \text{relation}(r) = \text{has relation} \}$. By injecting information from the KG into the data generation step, we ensure the generated samples are more contextually accurate and semantically rich.

**Topics $\mathcal{T}_{\text{LLM}}$ queried from Parametric LLMs.** Pre-trained on extensive text corpora such as medical literature, LLMs provide an alternative method for acquiring domain knowledge. Specifically, we aim to harness the rich clinical domain knowledge encoded in ChatGPT (gpt-3.5-turbo-0301) to augment the prompt. The incorporated prior knowledge from LLMs focus on entity types that hold relevance within clinical text datasets, including diseases, drugs, symptoms, and side effects. For each of entity types $e_i$, we prompt the LLMs by formulating inquiries $q(e_i)$, e.g., “Could you list 300 entities about <Entity Type>?”.

These crafted conversational cues serve as effective prompts to retrieve clinically significant entities from the rich domain knowledge within LLMs as $\mathcal{T}_{\text{LLM}} \sim \rho (\cdot; q(e_i))$. For each entity type, we generate 300 entities for synthetic data generation.

### 4.1.2 Clinical Writing Styles Suggestion

**Styles suggested by LLMs.** To address the limitations mentioned in Sec 3.2 and introduce a diverse range of writing styles $\mathcal{W}$ for synthetic samples, we leverage the powerful LLM to suggest candidate writing styles for each task. Specifically, for the task $i$, we incorporate task names $n_i$ into our prompts $p_i^{\text{style}}$ (e.g., disease entity recognition, recognizing text entailment) and integrate few-shot
with the generated topics and styles, the key challenge becomes how to leverage them to extract rich clinical information from the LLM for improving synthetic data quality. Directly putting all the elements to enrich the prompt is often infeasible due to the massive size of entities. To balance informativeness as well as diversity, we propose a knowledge-infused strategy, where for each class label name \( j \in \mathcal{Y} \), the collected clinical topics and writing styles serve as the base unit. In each step, we randomly sample a topic \( t \in \mathcal{T} \) and a writing style \( w \in \mathcal{W} \) from the candidate set to augment the prompt for class \( j \in \mathcal{Y} \) as \( p_j^{\text{Clin}}(t, w) = [p_j, t, w] \). Then, we use the augmented prompt \( p_j^{\text{Clin}}(t, w) \) together with the few-shot demonstrations \( \mathcal{D} \) to generate the synthetic dataset \( \mathcal{D}_{\text{Clin}}^{\text{Aug}} \):

\[
\tilde{\mathcal{D}}_{\text{Clin}} \sim p_j^{\text{Clin}}(t, w) \sim \mathcal{T}, w \sim \mathcal{W} \quad (\vdots [p_j, t, w], \mathcal{D})
\]

Despite its simplicity, this strategy enjoys several merits: (1) Clinical infusion: the clinical context is incorporated into the prompts to directly guide data generation; (2) Diversity: it encourages data diversity via dynamically composing different entities and writing styles into prompts; (3) Flexibility: it is compatible with different sources of \( \mathcal{T} \) and \( \mathcal{W} \) without reliance on specific knowledge formats. Consequently, the quality and clinical relevance of the generated synthetic data are enhanced. While some works focus on prompt optimization for data generation or other NLP tasks, they typically utilize a fixed prompt and optimize this prompt format, which is orthogonal to CLINGEN.

### 4.3 Language Model Fine-tuning

After generating synthetic data \( \tilde{\mathcal{D}} \), we fine-tune a pre-trained classifier \( C_0 \) for each downstream task. Following Meng et al. (2023), we first fine-tune \( C_0 \) on \( \mathcal{D} \) with standard supervised training objectives on few-shot examples (denoted as \( \ell(\cdot) \)) in Stage 1, then on synthetic data \( \tilde{\mathcal{D}} \) in Stage 2 as

\[
\theta^{(1)} = \min_{\theta} \mathbb{E}_{(x,y) \sim \mathcal{D}} \ell(f(x; \theta), y),
\]

\[
\theta^{(2)} = \min_{\theta} \mathbb{E}_{(\tilde{x},\tilde{y}) \sim \tilde{\mathcal{D}}} \ell(f(\tilde{x}; \theta), \tilde{y}), \theta_{\text{init}} = \theta^{(1)}.
\]

It’s important to highlight that we strictly follow a standard fine-tuning process and avoid using any extra techniques: (1) for standard classification tasks, \( \ell(\cdot) \) is the cross-entropy loss; (2) for multi-label classification tasks, \( \ell(\cdot) \) is the binary cross-entropy loss; (3) for token-level classification tasks, we stack an additional linear layer as the classification head and \( \ell(\cdot) \) is the token-level cross-entropy loss. The design of advanced learning objectives as well as data mixing strategies, while important, are orthogonal to the scope of this paper.

### 5 Empirical Evaluation

Given our focus on data generation, our major interest lies in faithfully evaluating different synthetic text generation approaches under few-shot scenarios, rather than competing in a “state-of-the-art” race with general few-shot NLP methods. The following questions particularly intrigue us: **RQ1**: How does CLINGEN perform when compared with baselines on different downstream tasks? **RQ2**: What impact do factors like LLM generators and synthetic data size have on the performance of CLINGEN? **RQ3**: How is the quality of the synthetic data generated by CLINGEN and baselines?  

5.1 Experiment Setup

We conduct experiments in the few-shot settings with 5 examples for each class. We employ ChatGPT (OpenAI, 2023b) (gpt-3.5-turbo-0301) as the LLM generator \( \mathcal{M} \) and maintain the same amount of synthetic training data for both CLINGEN and baselines for a fair comparison. The pre-trained PubMedBERT (Gu et al., 2021) is then applied to fine-tune on the synthetic data for both CLINGEN and baselines, where we consider both the Base and Large model.

**Datasets and Tasks.** We undertake a comprehensive evaluation of 18 datasets across a diverse array of tasks in clinical NLP benchmarks (Peng et al., 2019; Fries et al., 2022): 2 text classification, 3 relation extraction (RE), 3 natural language inference (NLI), 2 fact verification, 2 question answering (QA), 1 sentence similarity (STS), 4 Named Entity Recognition (NER), and 1 attribute extraction datasets. Please see Appendix C for descriptions and the statistics of each dataset.

**Baselines.** We compare CLINGEN with 10 baselines in total, including 6 data augmentation and...
We generate 5000 synthetic training data for both LLM-based data generation techniques. See Appendix D for their descriptions.

**Implementation Details.** For implementation, we use PyTorch (Paszke et al., 2019) and HuggingFace (Wolf et al., 2019). For each dataset, we randomly sample 5 examples from each class to provide few-shot demonstrations and keep a validation set of the same size. During the data generation process when we call the ChatGPT APIs (OpenAI, 2023b), we set the parameter $\text{top}_p = 1.0$ and temperature $t = 1.0$ to balance between the quality of the generated text as well as diversity (Chung et al., 2023; Yu et al., 2023)$^4$. In the experiments, we generate 5000 synthetic training data for both CLINGEN and the baselines and report the average performance over 3 random seeds for all the results. With the generated synthetic dataset, we follow the common few-shot learning setting (Perez et al., 2021) to train all the models for 6 epochs and use the model with the best performance on the validation set for evaluation. During the PubMedBERT fine-tuning, we adopt AdamW (Loshchilov and Hutter, 2019) for optimization with a linear warmup of the first 5% steps and linear learning rate decay. The learning rate is set to 2e-5 for Base and 1e-5 for Large, and the maximum number of tokens per sequence is 256.

---

### 5.2 Model Performance with Synthetic Data

Table 1 summarizes the experimental results. Due to space limits, we report the average performance over all datasets for each task, but provide the detailed results for each dataset in Tables 7, 8, 9 in Appendix F. Based on the experimental results, we have the following findings:

- Our approach, CLINGEN, consistently outperforms the baselines across all tasks. The average performance gain over all main metrics is 8.7% at Base scale and 7.7% at Large scale. LLM-based methods outperform traditional DA techniques, showcasing their ability to capture task-specific information from a few examples. DemoGen and ProGen’s gains over ZeroGen highlight the positive impact of few-shot examples. Despite being one of the most powerful data generation approaches,
S3’s gains are marginal in the few-shot setting due to its reliance on large validation sets.

- In token classification tasks, CLINGEN performs better with KG compared to LLM due to the better alignment between the task’s target and the generated domain knowledge, where the extracted topics serve as direct labels. Conversely, single-sentence and sentence-pair tasks favor LLM-based knowledge extraction. This could be because (1) These tasks prioritize sentence comprehension over specific terminologies, and some specialized terms might even impede LLM comprehension. (2) KGs may not always contain the required information, e.g., certain relations in chemical/protein relation extraction tasks, limiting performance gains.

- Some DA methods are task-specific, limiting their generalizability. For example, LightNER and KGPC are designed for NER. It is also non-trivial to apply Back Translation to NER or RE, as it requires locating related entities in the generated sentence accurately. In contrast, CLINGEN is flexible and can be readily applied to various tasks.

### 5.3 Ablation and Parameter Studies

#### Effect of Different LLM Generators

To investigate the impact of various LLMs on CLINGEN, we utilize InstructGPT (text-curie-001) (Ouyang et al., 2022) and GPT-4 (OpenAI, 2023a). Note that we only generate 500 samples in the GPT-4 setting due to budget constraints, but we provide the results of GPT-3.5 with same amount of synthetic samples for a fair comparison. From Figure 3 we observe that CLINGEN generally outperforms the best baseline in all settings. Additionally, we observe generally improved performance with larger models, as they often have better capabilities to follow our designed instructions for the given prompts. See Appendix G for more results.

**Effect of Size of Synthetic Data.** In Figure 4 (and more in Appendix G), we study the effect of the size of synthetic data. The result shows that CLINGEN consistently outperforms the best baseline, using only around 10% of the synthetic examples. This illustrates that incorporating domain knowledge and increasing the diversity of the prompts could be an effective way to improve the sample efficiency and narrow the gap between the performance of synthetic and ground-truth datasets.

**Comparison with few-shot inference via prompting LLM.** We also evaluate the performance of 5-shot in-context learning with ChatGPT and 3 medical LLMs, namely PMC-LLaMa-13b (Wu et al., 2023), MedAlpaca-13b (Han et al., 2023). Due to budget limits, we run experiments on datasets with few testing samples for each task. As presented in Table 2, CLINGEN at PubMedBERTLarge scale achieves better results on 5 out of 6 datasets than ChatGPT few-shot learning, which uses \( \sim 530 \times \) more parameters. One exception is for PUBHEALTH, as it requires complex reasoning abilities that PubMedBERTLarge may not fully possess. Three medical LLMs, on the other hand, perform less effectively than both CLINGEN and GPT-3.5 due to fewer parameters, limited reasoning capabilities, and training on a general medical corpus.
unsuited for the tasks. Overall, CLINGEN offers cost-effective and time-efficient advantages. While it entails a one-time investment in both money and time for synthetic training data generation, subsequent prediction relying on a moderate-sized model is much more efficient. Besides, the continued use of ChatGPT for inference on new testing data incurs ongoing time and financial costs, while our model requires zero additional costs for new data.

**Effect of Topic Extraction and Style Suggestion.** We inspect different components of CLINGEN in Table 3. It is observed that both Topics Extraction and Style Suggestion contribute to model performance as they enhance the relevance of generated samples to domain knowledge and introduce greater diversity. Different from the other datasets, MEDIQA-RQE shows more performance gain incorporating writing style than topics. It is because NLI tasks focus on capturing the relationships between two sentences while incorporating additional knowledge entities does not directly help the model improve the reasoning ability.

### 6 Quality Analysis of the Synthetic Data

**Data Distribution Measures.** Figure 5(a) shows the t-SNE plot of data generated by CLINGEN and baselines compared with the ground truth. This visualization demonstrates that CLINGEN exhibits a greater overlap with the ground truth, indicating a similar distribution as the original dataset. In addition, as depicted in Figure 6(a), the embedding of CLINGEN aligns more closely with the ground truth distribution than other baselines across all six datasets, further justifying the efficacy of CLINGEN for mitigating the distribution shift issue.

**Diversity Measures.** Table 4 calculates the average cosine similarity for sample pairs using Sentence-BERT embeddings. Compared to baselines, the dataset generated with CLINGEN exhibits lower cosine similarity and the average similarity is close to that of the ground truth training data, which shows CLINGEN could render more diverse data.

Moreover, Figure 6(b) highlights CLINGEN covers a broader range of entities than baselines, with CLINGEN w/ KG capturing more entities due to KGs’ extensive knowledge. Figure 6(c) reflects CLINGEN has a more balanced entity frequency distribution aligned with ground truth, ensuring diverse topic coverage.

**Case Study.** In Figure 5(b), we present a case study of generated examples.
study of examples generated by CLINGEN with LLM on MEDIQA-RQE dataset, which consists of consumer health queries. The examples reveal that the sentences generated by CLINGEN include more extensive contextual information compared with the baseline. These sentences closely resemble the queries people might pose in real-life scenarios.

**Study on Factual Consistency.** A human evaluation was carried out to assess the factual accuracy of the generated outputs across six representative tasks: LitCovid, CDR, Mediq-RQE, MQP, PubHealth, and BC5CDR. For each task, a sample of 100 examples per class was randomly selected. Medical students then examine the generated text and evaluate its factuality. The findings from this rigorous human study revealed no instances of misinformation or hallucinated content in the randomly sampled examples, verifying the system’s reliability in generating factually sound outputs.

**Monetary Cost** We display the monetary cost of CLINGEN for calling the OpenAI APIs, with a comparison with prompting GPT-3.5 for direct inference and DemoGen. From the values shown in Table 5, we observe that inference via GPT-3.5 generally has a higher cost, as it needs to input all the testing samples for prompting. In contrast, DemoGen has a relatively lower cost, because it does not include the topics and writing styles to the prompts as CLINGEN does.

### Table 5: The average cost (in US dollars) of running CLINGEN on various datasets per 1000 samples, compared with prompting GPT-3.5 for inference and DemoGen.

<table>
<thead>
<tr>
<th>Method</th>
<th>HOC</th>
<th>GAD</th>
<th>ChemProt</th>
<th>MEDIQA-RQE</th>
<th>PUBHEALTH</th>
<th>NCBI-Disease</th>
<th>CASI</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT-3.5 Inference</td>
<td>1.09</td>
<td>1.08</td>
<td>5.75</td>
<td>2.15</td>
<td>2.80</td>
<td>0.90</td>
<td>1.30</td>
</tr>
<tr>
<td>DemoGen</td>
<td>0.59</td>
<td>0.66</td>
<td>1.35</td>
<td>0.81</td>
<td>0.92</td>
<td>1.12</td>
<td>1.28</td>
</tr>
<tr>
<td>CLINGEN w/ KG</td>
<td>0.65</td>
<td>0.73</td>
<td>1.47</td>
<td>0.86</td>
<td>1.01</td>
<td>1.41</td>
<td>1.55</td>
</tr>
<tr>
<td>CLINGEN w/ LLM</td>
<td>0.72</td>
<td>0.84</td>
<td>1.51</td>
<td>0.90</td>
<td>1.34</td>
<td>1.49</td>
<td>1.62</td>
</tr>
</tbody>
</table>

In conclusion, we study clinical text data generation using LLMs. We thoroughly assess existing methods for clinical data generation and identify issues including distribution shifts and limited diversity. To tackle these challenges, we introduce CLINGEN, a framework that leverages clinical knowledge from non-parametric KGs and parametric LLMs. This empowers data generation by utilizing clinical topic knowledge and real-world writing styles in domain-specific prompts. Our extensive empirical evaluations across 8 clinical NLP tasks and 18 datasets, compared to 10 baseline methods, consistently show that CLINGEN improves task performance, aligns closely with real data, and enhances data diversity. We expect CLINGEN can be seamlessly incorporated into a broad suite of clinical text tasks to advance clinical NLP research.

### Acknowledgement

We thank the anonymous reviewers and area chairs for valuable feedbacks. This research was partially supported by the Emory Global Diabetes Center of the Woodruff Sciences Center, Emory University. Research reported in this publication was supported by the National Institute Of Diabetes And Digestive And Kidney Diseases of the National Institutes of Health under Award Number K25DK135913. The research also receives partial support by the National Science Foundation under Award Number IIS-2145411. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. We also thank Microsoft for providing research credits under the Accelerating Foundation Models Research Program.

### Limitation

In this work, we propose CLINGEN to better harness the LLM for synthetic text data generation. Despite its strong performance, we mainly verify their efficacy from their empirical performance, sample diversity, and distribution gaps. There are still some limitations to this work:

**Factuality of LLM-generated Text.** One issue with LLM-based synthetic data generation is the phenomenon of *hallucination*, wherein the model generates information that does not ground in reality (Zhang et al., 2023). This can lead to the propagation of misinformation, which may have negative impacts on the clinical domain. However, we have conducted a human study to justify that our generated synthetic data does not suffer from the issue of misinformation.

**Application to other type of clinical data.** Apart from text, there are other types of clinical data: For example, EHR data falls within a distinct
modality (i.e. tabular data) from textual data, which may require different methodologies and approaches (Wornow et al., 2023).

Ethics Consideration

On specific issue is about patient privacy. To eliminate this concern, we carefully select the five few-shot demonstrations to ensure they are fully free from any Protected Health Information (PHI) related to patients. We also make a deliberate effort to avoid any instructions that can potentially extract sensitive patient information within the prompts. Lastly, we conduct rigorous inspections of the generated synthetic data across all covered tasks to affirm that no such private information exists in the synthetic data generated by our method. In addition, we have opted out of human review for the data by completing the Azure OpenAI Additional Use Case Form5. This allows us to use the Azure OpenAI service while ensuring Microsoft does not have access to patient data.

References


Alex Bravo, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I Furlong. 2015. Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. BMC Bioinformatics, 16(1).


5https://aka.ms/oai/additionalusecase
Toronto, Canada. Association for Computational Linguistics.


OpenAI. 2023b. Introducing chatgpt.


George Tsatsaronis, Georgios Balikas, Prodomos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia


A Details on the Calculation of CMD

We introduce the Central Moment Discrepancy (CMD) (Zellinger et al., 2017), which is a widely used metric to measure the domain shift in the area of domain-invariant representation learning. Let \( X = (x_1, \ldots, x_n) \) and \( Y = (y_1, \ldots, y_n) \) be bounded feature vectors independent and identically distributed from two probability distributions \( p \) and \( q \). The central moment discrepancy metric (CMD) is defined by

\[
\text{CMD}(p, q) = \frac{1}{|b - a|} \left\| \mathbb{E}(X) - \mathbb{E}(Y) \right\|_2 + \sum_{k=2}^{\infty} \frac{1}{|b - a|^k} \left\| c_k(X) - c_k(Y) \right\|_2
\]

where \( \mathbb{E}(X) \) is the expectation of \( X \), and

\[
c_k(X) = \left( \mathbb{E} \left( \prod_{i=1}^{N} (X_i - \mathbb{E}(X_i))^r_i \right) \right)_{r_1+\cdots+r_N=k \atop r_1, \ldots, r_N \geq 0}
\]

is the central moment vector of order \( k \). To estimate the CMD efficiently without infinite-order calculation, we follow (Zellinger et al., 2017) and use a \( K \)-order approximation of CMD as

\[
\text{CMD}_K(p, q) = \frac{1}{|b - a|} \left\| \mathbb{E}(X) - \mathbb{E}(Y) \right\|_2 + \sum_{k=2}^{K} \frac{1}{|b - a|^k} \left\| C_k(X) - C_k(Y) \right\|_2
\]

where \( \mathbb{E}(X) = \frac{1}{|X|} \sum_{x \in X} x \) is the empirical expectation vector computed on the sample \( X \) and \( C_k(X) = \mathbb{E} \left( (x - \mathbb{E}(X))^k \right) \) is the vector of all \( k \)-th order sample central moments of the coordinates of \( X \). To adapt CMD in our work, we set \( K = 5 \), and use the embedding from SentenceBERT (Reimers and Gurevych, 2019) to calculate the embedding \( X, Y \) for instances.

B Additional Preliminary Studies

We present additional preliminary studies of the t-SNE plots in Figure 7 and the regularized entity frequencies in Figure 8. In Figure 7, we visualize the embeddings\(^7\) of both the ground truth training data and synthetic datasets generated via two representative methods. Overall, these methods use generic prompts (see Appendix E.3 for details) with minimal domain-specific constraints. These results further justify the distribution shift issue mentioned in section 3.2, demonstrating that the limited diversity as well as the distribution shift issue generally exists for a broad range of clinical NLP tasks.

Figure 9 shows a case study, where we randomly select one sample from each class within the training set generated by ZeroGen and DemoGen. These selected samples are compared with the ground truth data from the MEDIQA-RQE dataset, which aims to predict whether a consumer health query can entail an existing Frequently Asked Question (FAQ). It is evident that the samples generated by ZeroGen and DemoGen exhibit a limited range of writing styles and tend to follow a specific template, whereas the ground truth sample contains more contextual elements that are typically encountered in real-life scenarios.

C Dataset Description

The evaluation tasks and datasets are summarized in Table 6. Note that the number of training samples indicates the size of the original training set. Specifically, we consider the following datasets:

- **Single-Sentence Tasks**
  - Text Classification:
    - The LitCovid dataset (Chen et al., 2021) consists of COVID-19-related publications from PubMed. The task is to predict the topics of the sentences, including “Epidemic Forecasting”, “Treatment”, “Prevention”, “Mechanism”, “Case Report”, “Transmission”, and “Diagnosis”.
    - The HOC dataset (Baker et al., 2015) also extracts sentences from PubMed articles, each annotated at the sentence level. The task is to predict the topics of the sentences, including “evading growth suppressors”, “tumor promoting inflammation”, “enabling replicative immortality”, “cellular energetics”, “resisting cell death”, “activating invasion and metastasis”, “genomic instability and mutation”, “inducing angiogenesis”, “sustaining proliferative signaling”, and “avoiding immune destruction”.
  - Relation Extraction:

\(^6\)The implementation of CMD is available at https://gist.github.com/yusuke0519/724aa68fc431afad8b0cc72801568da17b
\(^7\)We employ SentenceBERT (Reimers and Gurevych, 2019) as the text encoder.
Figure 7: The t-SNE plots of datasets generated by ZeroGen and DemoGen compared with the ground truth.

- The GAD (Bravo et al., 2015) dataset is to predict whether there is a relation between the given disease and gene in the sentences. Note that the original annotation for this dataset is Noisy. To remedy this issue, we relabel 350 examples from the original test set to form a clean subset for faithful evaluation.
- The CDR (Wei et al., 2016) dataset is to predict whether the provided chemical can induce the disease in the sentences.
- The ChemProt (Taboureau et al., 2010) dataset focuses on the chemical-protein relations, and the labels include “Upregulator”, “Downregulator”, “Agonist”, “Antagonist”, “Product_of” and “No relation”.

• Sentence-Pair Tasks
  - Natural Language Inference (NLI):
    - The MedNLI (Shivade, 2017) dataset consists of sentences pairs derived from MIMIC-III, where we predict the relations between the sentences. The labels include “entailment”, “neutral” and “contradiction”.
  - The MEDIQA-NLI (Ben Abacha et al., 2019) dataset comprises text-hypothesis pairs. Their relations include “entailment”, “neutral” and “contradiction”.
  - The MEDIQA-RQE (Abacha and Demner-Fushman, 2016) dataset contains NIH consumer health question pairs, and the task is to recognize if the first question can entail the second one.
  - Fact Verification:
    - The PUBHEALTH (Kotonya and Toni, 2020) encompasses claims paired with journalist-crafted explanations. The task is to predict the relations between the claim and evidence, including “Refute”, “Unproven”, “Support”, and “Mixture”.
    - The HealthVer (Sarrouti et al., 2021) contains evidence-claim pairs from search engine snippets regarding COVID-19 questions. The relations between claims and evidences are chosen from “Refute”, “Unproven”, and “Support”.
  - Question Answering (QA):
    - The PubmedQA task (Jin et al., 2019) en-
Figure 8: The regularized entity frequencies of datasets generated by ZeroGen and DemoGen compared with the ground truth in log scale.

Figure 9: Case study of generated samples by existing methods ZeroGen and DemoGen.

tails responding to inquiries regarding the abstracts of biomedical research papers.

- The BioASQ task (Tsatsaronis et al., 2015) spans multiple question types, including factoid, list, summary, and yes/no questions derived from expert-reviewed biomedical research papers.

- Sentence Similarity (STS):
  - the MQP (McCreery et al., 2020) dataset comprises a collection of medical question pairs designed for identifying semantically similar questions. The task is to predict whether the two questions are equivalent or not.

- Token Classification Tasks
  - Named Entity Recognition (NER):
    - The BC5CDR-Disease (Li et al., 2016) is to recognize diseases in the sentences.
    - The BC5CDR-Chemical (Li et al., 2016) is to recognize chemicals in the sentences.
  - The NCBI-Disease (Dogan et al., 2014) is to recognize diseases in the sentences.
<table>
<thead>
<tr>
<th>Corpus</th>
<th>Tasks</th>
<th>#Class</th>
<th>#Train/#Test</th>
<th>Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>LitCovid (Chen et al., 2021)</td>
<td>Text Classification</td>
<td>7</td>
<td>24960/6238</td>
<td>F1</td>
</tr>
<tr>
<td>HOC (Baker et al., 2015)</td>
<td>Text Classification</td>
<td>10</td>
<td>3091/898</td>
<td>F1</td>
</tr>
<tr>
<td>GAD (Bravo et al., 2015)</td>
<td>Relation Extraction (RE)</td>
<td>1</td>
<td>4750/350</td>
<td>P, R, F1</td>
</tr>
<tr>
<td>CDR (Wei et al., 2016)</td>
<td>Relation Extraction (RE)</td>
<td>1</td>
<td>8431/2522</td>
<td>P, R, F1</td>
</tr>
<tr>
<td>ChemProt (Taboureau et al., 2010)</td>
<td>Relation Extraction (RE)</td>
<td>5</td>
<td>8793/1087</td>
<td>F1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sentence-Pair Tasks</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MedNLI* (Shivade, 2017)</td>
<td>Natural Language Inference (NLI)</td>
<td>3</td>
<td>11232/1422</td>
<td>Acc</td>
</tr>
<tr>
<td>MEDIQA-NLI† (Ben Abacha et al., 2019)</td>
<td>Natural Language Inference (NLI)</td>
<td>3</td>
<td>-/405</td>
<td>Acc</td>
</tr>
<tr>
<td>MEDIQA-RQE (Abacha and Demner-Fushman, 2016)</td>
<td>Natural Language Inference (NLI)</td>
<td>2</td>
<td>8588/302</td>
<td>Acc</td>
</tr>
<tr>
<td>PUBHEALTH (Kotonya and Toni, 2020)</td>
<td>Fact Verification</td>
<td>4</td>
<td>9804/1231</td>
<td>Acc, F1</td>
</tr>
<tr>
<td>HealthVer (Sarroui et al., 2021)</td>
<td>Fact Verification</td>
<td>3</td>
<td>10591/1824</td>
<td>Acc, F1</td>
</tr>
<tr>
<td>MQP (McCreery et al., 2020)</td>
<td>Sentences Similarity (STS)</td>
<td>2</td>
<td>10/3033</td>
<td>Acc</td>
</tr>
<tr>
<td>PubmedQA (Jin et al., 2019)</td>
<td>Question Answering (QA)</td>
<td>2</td>
<td>500/500</td>
<td>Acc</td>
</tr>
<tr>
<td>BioASQ (Tsatsaronis et al., 2015)</td>
<td>Question Answering (QA)</td>
<td>2</td>
<td>670/140</td>
<td>Acc</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Token Classification Tasks</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>BCSCDR-Disease (Li et al., 2016)</td>
<td>Named Entity Recognition (NER)</td>
<td>1</td>
<td>4882/5085</td>
<td>P, R, F1</td>
</tr>
<tr>
<td>BCSCDR-Chemical (Li et al., 2016)</td>
<td>Named Entity Recognition (NER)</td>
<td>1</td>
<td>4882/5085</td>
<td>P, R, F1</td>
</tr>
<tr>
<td>NCBI-Disease (Dogan et al., 2014)</td>
<td>Named Entity Recognition (NER)</td>
<td>1</td>
<td>5336/921</td>
<td>P, R, F1</td>
</tr>
<tr>
<td>CHEMDNER (Krallinger et al., 2015)</td>
<td>Named Entity Recognition (NER)</td>
<td>1</td>
<td>14522/12430</td>
<td>P, R, F1</td>
</tr>
<tr>
<td>CASI (Agrawal et al., 2022; Moon et al., 2014)</td>
<td>Attribute Extraction</td>
<td>6</td>
<td>5/100</td>
<td>F1</td>
</tr>
</tbody>
</table>

Table 6: Dataset statistics. We do not count the non-entity/non-relation class for relation extraction and token classification tasks to align with existing works. P and R stand for Precision and Recall. Metrics in bold are considered as the main metrics. * is not allowed to put into GPT and † does not provide training data, so we sample few-shot examples from the SciTail (Khot et al., 2018) instead.

- The CHEMDNER (Krallinger et al., 2015) is to recognize chemicals in the sentences.
- Attribute Extraction (MedAttr):
  - The CASI dataset (Agrawal et al., 2022; Moon et al., 2014) aims to identify interventions including medication, dosage, route, freq, reason, duration

### D  Baseline Details

In this section, we give a detailed introduction for all baselines used in this study.

#### Data Augmentation Methods:

- **DA-Word Sub** (Ribeiro et al., 2020): It performs word substitution for few-shot demonstrations to create new training sample. Specifically, we follow Checklist (Ribeiro et al., 2020) and maintain a word list to generate new examples.

- **DA-Back Translation** (Xie et al., 2020): It employ back translation to augment the training data (Xie et al., 2020), including translating text from the target language to the source language and then back to the target language.

- **DA-Mixup** (Chen et al., 2020; Zhang et al., 2020): It adds interpolation on the embedding space of the training examples to create virtual augmented examples.

- **DA-Transformer** (MELM) (Kumar et al., 2020; Zhou et al., 2022): It introduces a conditional data augmentation technique that prepends class labels to text sequences for pre-trained transformer-based models. Specifically, it leverage the sequence to sequence transformer to perform conditional text generation based on the seed examples.

- **LightNER** (Chen et al., 2022a): It adopts a seq2seq framework, generating the entity span sequence and entity categories under the guidance of a self-attention-based prompting module. It is designed specifically for NER tasks.

- **KGPC** (Chen et al., 2023): It injects the semantic relations of the knowledge graph to sequence to text generation models to perform knowledge-guided instance generation for few-shot biomedical NER. It also only applies to NER tasks.

#### LLM-based Generation Methods.

- **ZeroGen** (Ye et al., 2022a): It generates a dataset using simple class-conditional prompts and then trains a tiny task-specific model for zero-shot inference. We follow the prompting
method mentioned in their original paper as implementation, which does not consider any style information as well as domain knowledge.

- **DemoGen** (Meng et al., 2023; Yoo et al., 2021): It leverages LLMs to synthesize novel training data by feeding few-shot samples as demonstrations to guide the data generation process. Note that we focus on using the black-box LLM as the generator, thus we do not tune the LLM as (Meng et al., 2023).

- **ProGen** (Ye et al., 2022b): It first identifies the most important examples from the generated synthetic data using the influence function, then adds these examples as demonstrations to generate new training instances. To ensure fair comparison, we also add the few-shot demonstrations for data generation.

- **S3** (Wang et al., 2023): It is a synthetic data generation method that iteratively extrapolates errors made by the classifier model trained on synthetic data leveraging a large language model. To adapt it in our few-shot setting, we use few-shot demonstrations $D$ as the validation set.

### E Prompt Format

#### E.1 The prompts for Writing Styles Suggestion with CLINGEN

Listing 1: Prompt Format for writing styles suggestion with CLINGEN.

Suppose you need to generate a synthetic clinical text dataset on [task] tasks. Here are a few examples from the original training set: [demonstrations]. Please write three potential sources, speakers or authors of the sentences.

(task): The task names for each specific task. 
[demonstrations]: The few-shot demonstrations from the original training set.

#### E.2 The prompts for Data Generation with CLINGEN

In the following prompt format, [topic] and [style] are randomly sampled from the topics candidate set and styles candidate set we formulate in the knowledge extraction step, respectively.

### Named entity recognition tasks:

Listing 2: Prompt Format for NER tasks with CLINGEN.

Suppose you need to create a dataset for [domain] recognition. Your task is to:
1. generate a sentence about [domain],
2. output a list of named entity about [domain] only,
3. the sentence should mimic the style of [style],
4. the sentence should mention the [domain] named [topic].

[domain]: “disease” for BC5CDR-Disease and NCBI-Disease; “chemical” for BC5CDR-Chemical and CHEMDNER.

### Medication attributes tasks:

Listing 3: Prompt Format for medication attributes tasks with CLINGEN.

Suppose you need to create a dataset for clinical attributes recognition. Your task is to:
1. generate a sentence about clinical attributes, The Clinical Attributes you need to extract include "Medication", "Dosage", "Route", "Frequency", "Reason", "Duration". For each attribute class, please return a list of attributes within the class that occurs in the Sentence.
2. the sentence should mimic the style of [style],
3. the sentence should be relevant to [topic].

### Text classification tasks:

Listing 4: Prompt Format for text classification tasks with CLINGEN.

Suppose you need to create a dataset for [domain]. Your task is to:
1. generate a sentence about [domain],
2. the sentence should mimic the style of [style],
3. the sentence should be relevant to the subtopic of [topic] for [class_name].
Relation extraction tasks:

Listing 5: Prompt Format for relation extraction tasks with CLINGEN.

Suppose you need to generate synthetic data for the biomedical [domain] task. Your task is to:
1. give a sentence about [class_name] relation between [entity0] and [entity1]
2. the sentence should discuss the [entity0]: [topic0] and [entity1]: [topic1] with the relation [label_desc].
3. the sentence should mimic the style of [style].

Natural language inference tasks:

Listing 6: Prompt Format for generating the first sentence in NLI tasks with CLINGEN.

Suppose you need to create a set of [content]. Your task is to:
1. generate one sentence for a [content].
2. the [content] should be relevant to [topic].
3. The [content] should mimic the style of [style].

E.3 Prompts for ZeroGen, DemoGen, ProGen

We use the same set of prompts for ZeroGen, DemoGen and ProGen, while DemoGen and ProGen have additional demonstrations augmented to the prompts. DemoGen uses the few-shot examples in the training set as demonstrations, and ProGen leverages feedbacks from previous rounds to iteratively guide the generation.

Named entity recognition tasks:

Listing 8: Prompt Format for NER tasks with baselines.

Suppose you need to create a dataset for [domain] recognition. Your task is to generate a sentence about [domain] and output a list of named entity about [domain] only.

Medication attributes tasks:

Listing 9: Prompt Format for medication attributes tasks with baselines.
Suppose you need to create a dataset for clinical attributes recognition. Your task is to generate a sentence about clinical attributes, The Clinical Attributes you need to extract include "Medication", "Dosage", "Route", "Frequency", "Reason", "Duration". For each attribute class, please return a list of attributes within the class that occurs in the sentence.

Text classification tasks:
Listing 10: Prompt Format for text classification tasks with baselines.

Suppose you are a writer for [domain]. Your task is to give a synthetic [domain] about [class_name].

[domain]: “COVID-19 Literature” for LitCovid and “Cancer Document” for HOC.
[class_name]: the label name for this generated sample, listed in Appendix C.

Relation extraction tasks:
Listing 11: Prompt Format for relation extraction tasks with baselines.

Suppose you need to generate synthetic data for the biomedical [domain] task. Your task is to give a sentence about [class_name] relation between [entity0] and [entity1] so that [label_desc].

[domain]: “Disease Gene Relation” for GAD, “Chemical Disease Relation” for CDR, and “Chemical Protein Relation” for ChemProt.
[entity0] and [entity1]: “disease” and “gene” for GAD, “chemical” and “disease: for CDR, and “chemical” and “protein” for ChemProt.
[class_name]: the label name for this generated sample, listed in Appendix C.
[label_desc]: the description of the selected label. For example, the label “upregulator” in ChemProt has a description of “the chemical activates expression of the protein.”

Natural language inference tasks:
Listing 12: Prompt Format for generating the first sentence in NLI tasks with baselines.

Suppose you need to create a set of [content]. Your task is to generate one sentence for a [content].

[content]: “health question” for MEDIQA-RQE, “claim” for MEDIQA-NLI, MedNLI and MQP, and “health news” for PUBHEALTH and HealthVer.

Listing 13: Prompt Format for generating the second sentence in NLI tasks with baselines.

Suppose you need to create a pair of sentences for the [domain] task with the label '[class_name]'. Given the [content]: [first_sentence], Your task is to generate one short [content] so that [label_desc].

[domain]: “Question Entailment” for MEDIQA-RQE, “Natural Language Entailment” for MEDIQA-NLI and MedNLI, “Fact Verification” for PUBHEALTH and HealthVer, and “Sentence Similarity Calculation” for MQP.
[content]: “health question” for MEDIQA-RQE, “hypothesis” for MEDIQA-NLI, MedNLI, “evidence” for PUBHEALTH and HealthVer, and “sentence” for MQP.
[class_name]: the label name for this generated sample, listed in Appendix C.
[label_desc]: the description of the selected label. For "entailment", the description is "we can infer the [content] from the given sentence". For "neutral", the description is "there is no clear relation between the [content] from the given sentence". For "contradict", the description is "we can refute the [content] from the given sentence".
[first_sentence]: the first sentence we generate.

F Detailed Per-task Experimental Results
In this section, we present additional experimental results on every dataset in Tables 7, 8, 9. We also include the experimental results combining topic from both KG and LLM, which yields a performance improvement, though not a substantial one. However, note that in practice, it is challenging to tune the ratio in the few-shot setting.
<table>
<thead>
<tr>
<th></th>
<th>LitCovid</th>
<th>HOC</th>
<th>CDR</th>
<th>GAD</th>
<th>ChemProt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>F1</td>
<td>P</td>
<td>R</td>
<td>F1</td>
</tr>
<tr>
<td><strong>PubMedBERT\textsubscript{Base}</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supervised-Full (SOTA)</td>
<td>73.55</td>
<td>84.35</td>
<td>67.81</td>
<td>76.60</td>
<td>71.96</td>
</tr>
<tr>
<td>Supervised-Full</td>
<td>71.70</td>
<td>82.32</td>
<td>67.81</td>
<td>76.60</td>
<td>71.96</td>
</tr>
<tr>
<td>Supervised-Few</td>
<td>24.08</td>
<td>13.13</td>
<td>41.62</td>
<td>52.96</td>
<td>46.61</td>
</tr>
<tr>
<td>DA-Word Sub</td>
<td>36.49</td>
<td>44.98</td>
<td>40.50</td>
<td>46.20</td>
<td>43.16</td>
</tr>
<tr>
<td>DA-Back Trans</td>
<td>39.70</td>
<td>54.78</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>DA-Mixup</td>
<td>40.82</td>
<td>49.35</td>
<td>41.40</td>
<td>44.80</td>
<td>43.03</td>
</tr>
<tr>
<td>DA-Transformer</td>
<td>39.86</td>
<td>42.18</td>
<td>44.60</td>
<td>61.70</td>
<td>51.77</td>
</tr>
<tr>
<td>ZeroGen</td>
<td>50.50</td>
<td>67.90</td>
<td>38.82</td>
<td>41.82</td>
<td>54.57</td>
</tr>
<tr>
<td>DemoGen</td>
<td>57.65</td>
<td>70.52</td>
<td>46.90</td>
<td>83.3</td>
<td>60.01</td>
</tr>
<tr>
<td>ProGen</td>
<td>58.06</td>
<td>72.25</td>
<td>51.35</td>
<td>71.58</td>
<td>59.80</td>
</tr>
<tr>
<td>S3</td>
<td>58.67</td>
<td>71.58</td>
<td>49.76</td>
<td>76.08</td>
<td>51.77</td>
</tr>
<tr>
<td><strong>CLINGEN w/ KG</strong></td>
<td>58.01</td>
<td>76.28</td>
<td>56.98</td>
<td>67.38</td>
<td>61.75</td>
</tr>
<tr>
<td><strong>CLINGEN w/ LLM</strong></td>
<td>59.22</td>
<td>76.42</td>
<td>60.60</td>
<td>66.35</td>
<td>63.34</td>
</tr>
<tr>
<td><strong>CLINGEN w/ KG+LLM</strong></td>
<td>56.56</td>
<td>78.02</td>
<td>57.97</td>
<td>71.09</td>
<td>63.86</td>
</tr>
<tr>
<td><strong>PubMedBERT\textsubscript{Large}</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supervised-Full (SOTA)</td>
<td>—</td>
<td>84.87</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Supervised-Full</td>
<td>74.59</td>
<td>85.53</td>
<td>72.31</td>
<td>74.88</td>
<td>73.57</td>
</tr>
<tr>
<td>Supervised-Few</td>
<td>22.59</td>
<td>13.13</td>
<td>42.27</td>
<td>67.51</td>
<td>51.99</td>
</tr>
<tr>
<td>DA-Word Sub</td>
<td>37.20</td>
<td>50.78</td>
<td>47.70</td>
<td>43.50</td>
<td>45.50</td>
</tr>
<tr>
<td>DA-Back Trans</td>
<td>40.50</td>
<td>61.46</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>DA-Mixup</td>
<td>40.03</td>
<td>53.45</td>
<td>43.34</td>
<td>73.50</td>
<td>54.37</td>
</tr>
<tr>
<td>DA-Transformer</td>
<td>38.95</td>
<td>49.86</td>
<td>50.70</td>
<td>31.60</td>
<td>38.93</td>
</tr>
<tr>
<td>ZeroGen</td>
<td>52.86</td>
<td>70.16</td>
<td>42.95</td>
<td>80.67</td>
<td>56.06</td>
</tr>
<tr>
<td>DemoGen</td>
<td>56.29</td>
<td>73.65</td>
<td>50.86</td>
<td>74.30</td>
<td>60.39</td>
</tr>
<tr>
<td>ProGen</td>
<td>54.71</td>
<td>75.31</td>
<td>50.36</td>
<td>76.08</td>
<td>60.60</td>
</tr>
<tr>
<td>S3</td>
<td>53.56</td>
<td>75.11</td>
<td>51.51</td>
<td>78.30</td>
<td>62.14</td>
</tr>
<tr>
<td><strong>CLINGEN w/ KG</strong></td>
<td>55.81</td>
<td>77.71</td>
<td>60.45</td>
<td>65.04</td>
<td>62.66</td>
</tr>
<tr>
<td><strong>CLINGEN w/ LLM</strong></td>
<td>57.07</td>
<td>78.14</td>
<td>67.13</td>
<td>62.98</td>
<td>64.99</td>
</tr>
<tr>
<td><strong>CLINGEN w/ KG+LLM</strong></td>
<td>56.80</td>
<td>79.07</td>
<td>64.19</td>
<td>67.70</td>
<td>65.90</td>
</tr>
</tbody>
</table>

Table 7: Performance on single-sentence tasks evaluated by PubMedBERT\textsubscript{Base} and PubMedBERT\textsubscript{Large}. **Bold** and underline indicate the best and second best results for each dataset, respectively. Note that the performance of 'Supervised-Full (SOTA)' is copied from the existing paper. If the value in this field is missing, this means we cannot find reported results with the same-scale model on that dataset. (Same as below).
<table>
<thead>
<tr>
<th></th>
<th>MEDIQA-RQE</th>
<th>MEDIQA-NLI</th>
<th>MedNLI</th>
<th>PUBHEALTH</th>
<th>HealthVer</th>
<th>MQP</th>
<th>PubmedQA</th>
<th>BioASQ</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ACC</td>
<td>ACC</td>
<td>ACC</td>
<td>ACC</td>
<td>F1</td>
<td>ACC</td>
<td>ACC</td>
<td>ACC</td>
</tr>
<tr>
<td>PubMedBERTbase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supervised-Full (SOTA)</td>
<td>—</td>
<td>—</td>
<td>86.60</td>
<td>70.52</td>
<td>69.73</td>
<td>73.54</td>
<td>74.82</td>
<td>79.20</td>
</tr>
<tr>
<td>Supervised-Full</td>
<td>77.15</td>
<td>79.01</td>
<td>81.43</td>
<td>65.16</td>
<td>62.96</td>
<td>70.00</td>
<td>68.02</td>
<td>75.70</td>
</tr>
<tr>
<td>Supervised-Few</td>
<td>57.51</td>
<td>40.00</td>
<td>36.40</td>
<td>28.30</td>
<td>23.70</td>
<td>30.55</td>
<td>30.49</td>
<td>55.70</td>
</tr>
<tr>
<td>DA-Word Sub</td>
<td>58.60</td>
<td>50.24</td>
<td>56.40</td>
<td>23.67</td>
<td>17.64</td>
<td>34.05</td>
<td>34.02</td>
<td>54.40</td>
</tr>
<tr>
<td>DA-Back Trans</td>
<td>59.16</td>
<td>49.92</td>
<td>53.82</td>
<td>30.70</td>
<td>23.32</td>
<td>33.60</td>
<td>32.76</td>
<td>55.80</td>
</tr>
<tr>
<td>DA-Mixup</td>
<td>57.71</td>
<td>49.38</td>
<td>53.47</td>
<td>31.45</td>
<td>24.45</td>
<td>34.11</td>
<td>33.78</td>
<td>58.20</td>
</tr>
<tr>
<td>DA-Transformer</td>
<td>62.25</td>
<td>51.19</td>
<td>53.70</td>
<td>34.81</td>
<td>27.75</td>
<td>35.83</td>
<td>35.78</td>
<td>58.80</td>
</tr>
<tr>
<td>ZeroGen</td>
<td>63.28</td>
<td>52.89</td>
<td>57.71</td>
<td>35.80</td>
<td>31.50</td>
<td>34.80</td>
<td>33.50</td>
<td>68.35</td>
</tr>
<tr>
<td>DemoGen</td>
<td>66.56</td>
<td>56.29</td>
<td>58.56</td>
<td>42.60</td>
<td>35.40</td>
<td>38.00</td>
<td>36.50</td>
<td>70.85</td>
</tr>
<tr>
<td>ProGen</td>
<td>65.94</td>
<td>57.28</td>
<td>59.49</td>
<td>38.70</td>
<td>33.10</td>
<td>36.72</td>
<td>35.97</td>
<td>69.30</td>
</tr>
<tr>
<td>S3</td>
<td>66.02</td>
<td>58.30</td>
<td>59.75</td>
<td>42.40</td>
<td>34.90</td>
<td>37.94</td>
<td>37.97</td>
<td>70.20</td>
</tr>
<tr>
<td>CLINGEN w/ KG</td>
<td>74.85</td>
<td>58.03</td>
<td>61.80</td>
<td>44.60</td>
<td>36.80</td>
<td>43.05</td>
<td>42.06</td>
<td>72.20</td>
</tr>
<tr>
<td>CLINGEN w/ LLM</td>
<td>72.40</td>
<td>64.44</td>
<td>64.89</td>
<td>48.50</td>
<td>40.60</td>
<td>44.50</td>
<td>42.32</td>
<td>73.30</td>
</tr>
<tr>
<td>CLINGEN w/ KG+LLM</td>
<td>75.10</td>
<td>64.12</td>
<td>65.81</td>
<td>50.57</td>
<td>40.65</td>
<td>40.60</td>
<td>39.59</td>
<td>68.30</td>
</tr>
</tbody>
</table>

|              | PubMedBERTbase |            |        |           |           |     |          |        |
|              |              |            |        |           |           |     |          |        |
| Supervised-Full (SOTA) | —          | —          | 86.57  | —         | —         | —   | 81.00    | 72.18  | 94.82  |
| Supervised-Full     | 81.10        | 82.89      | 83.96  | 70.21     | 63.45     | 75.72| 75.01    | 78.80  | 67.38  | 93.36  |
| Supervised-Few      | 63.79        | 47.40      | 38.80  | 46.20     | 27.20     | 35.60| 33.80    | 59.73  | 60.44  | 58.57  |
| DA-Word Sub         | 64.26        | 51.20      | 57.53  | 35.60     | 31.60     | 35.41| 32.29    | 55.30  | 55.72  | 61.42  |
| DA-Back Trans       | 65.52        | 51.43      | 58.21  | 34.45     | 30.50     | 33.78| 32.21    | 56.40  | 54.38  | 60.00  |
| DA-Mixup            | 64.10        | 50.91      | 57.03  | 34.23     | 30.78     | 33.79| 31.42    | 58.50  | 54.80  | 58.57  |
| DA-Transformer      | 68.97        | 51.05      | 56.79  | 38.46     | 31.40     | 37.94| 37.97    | 70.20  | 58.60  | 60.00  |
| ZeroGen             | 67.26        | 60.74      | 62.42  | 42.50     | 33.30     | 39.74| 38.90    | 72.69  | 57.75  | 74.28  |
| DemoGen             | 69.22        | 62.97      | 64.55  | 44.50     | 36.80     | 40.72| 40.57    | 74.37  | 61.50  | 68.57  |
| ProGen              | 67.82        | 60.98      | 63.15  | 44.15     | 36.37     | 41.42| 40.89    | 74.90  | 59.40  | 67.14  |
| S3                  | 67.98        | 63.15      | 64.10  | 43.72     | 35.67     | 39.80| 39.78    | 73.20  | 61.20  | 71.42  |
| CLINGEN w/ KG       | 79.92        | 63.59      | 69.19  | 50.20     | 41.26     | 47.03| 43.64    | 75.40  | 68.60  | 79.28  |
| CLINGEN w/ LLM      | 77.26        | 64.69      | 69.46  | 52.96     | 43.31     | 46.05| 44.12    | 76.20  | 66.80  | 80.80  |
| CLINGEN w/ KG+LLM   | 80.77        | 63.30      | 70.56  | 51.98     | 41.61     | 47.44| 44.25    | 71.90  | 67.40  | 79.28  |

Table 8: Performance on sentence-pair tasks evaluated by PubMedBERTbase and PubMedBERTlarge.
<table>
<thead>
<tr>
<th></th>
<th>BC5CDR-Disease</th>
<th>BC5CDR-Chemical</th>
<th>NCBI-Disease</th>
<th>CHEMDNER</th>
<th>CASI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>F1</td>
<td>P</td>
<td>R</td>
</tr>
<tr>
<td>PubMedBERTbase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supervised-Full (SOTA)</td>
<td>---</td>
<td>---</td>
<td>86.10</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Supervised-Full</td>
<td>83.84</td>
<td>87.92</td>
<td>85.83</td>
<td>92.22</td>
<td>91.74</td>
</tr>
<tr>
<td>Supervised-Few</td>
<td>24.86</td>
<td>39.47</td>
<td>30.51</td>
<td>63.73</td>
<td>46.07</td>
</tr>
<tr>
<td>DA-Weed Sub</td>
<td>35.34</td>
<td>39.54</td>
<td>37.32</td>
<td>63.13</td>
<td>52.52</td>
</tr>
<tr>
<td>DA-Mixup</td>
<td>36.13</td>
<td>42.90</td>
<td>39.23</td>
<td>66.43</td>
<td>50.54</td>
</tr>
<tr>
<td>LightNER</td>
<td>39.80</td>
<td>33.20</td>
<td>36.20</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>DA-MELM</td>
<td>34.20</td>
<td>41.30</td>
<td>37.42</td>
<td>47.23</td>
<td>72.81</td>
</tr>
<tr>
<td>KGPC</td>
<td>50.80</td>
<td>51.30</td>
<td>51.05</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>ZeroGen</td>
<td>55.60</td>
<td>39.10</td>
<td>45.91</td>
<td>73.20</td>
<td>82.85</td>
</tr>
<tr>
<td>DemoGen</td>
<td>63.10</td>
<td>48.44</td>
<td>54.81</td>
<td>76.40</td>
<td>81.65</td>
</tr>
<tr>
<td>ProGen</td>
<td>61.60</td>
<td>50.50</td>
<td>55.50</td>
<td>77.10</td>
<td>82.02</td>
</tr>
<tr>
<td>S3</td>
<td>58.26</td>
<td>55.96</td>
<td>57.08</td>
<td>77.28</td>
<td>80.80</td>
</tr>
<tr>
<td>CLINGEN w/ KG</td>
<td>58.64</td>
<td>63.02</td>
<td>60.75</td>
<td>74.96</td>
<td>85.45</td>
</tr>
<tr>
<td>CLINGEN w/ LLM</td>
<td>63.41</td>
<td>58.83</td>
<td>61.03</td>
<td>77.68</td>
<td>84.33</td>
</tr>
</tbody>
</table>

Table 9: Performance on token-classification tasks evaluated by PubMedBERTbase and PubMedBERTlarge.

Figure 10: Different generators at Base.

Figure 11: Different proportion of data at Base.

<table>
<thead>
<tr>
<th></th>
<th>HOC</th>
<th>CDR</th>
<th>MEDRQA-RQE</th>
<th>NCBIDisease</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best Baseline</td>
<td>CLINGEN-KG</td>
<td>CLINGEN-LLM</td>
<td>Best Baseline</td>
</tr>
<tr>
<td>1</td>
<td>74.30</td>
<td>77.30</td>
<td>74.30</td>
<td>77.30</td>
</tr>
<tr>
<td>2</td>
<td>74.30</td>
<td>77.30</td>
<td>74.30</td>
<td>77.30</td>
</tr>
<tr>
<td>3</td>
<td>74.30</td>
<td>77.30</td>
<td>74.30</td>
<td>77.30</td>
</tr>
</tbody>
</table>

Table 10: Performance with Different Random Seeds using PubMedBERTbase.
Figure 12: The t-SNE plots of datasets generated by CLINGEN, ZeroGen and DemoGen compared with the ground truth.

Figure 13: The regularized entity frequencies of datasets generated by CLINGEN, ZeroGen and DemoGen compared with the ground truth in log scale.
G Additional Ablation and Parameter Studies

Figure 10 and 11 show the effect of different generators and the effect of the proportion of data on two additional datasets, respectively. Overall, our method generally outperforms the best baseline. One interesting finding for the NCBI-Disease dataset is that CLINGEN performs worse than the best on one variant. We hypothesize that it is because this task involves more complex input and output, potentially posing a challenge for moderate-size LLMs to follow the instructions.

Besides, as few-shot sample selection is important for the final performance, we show the performance of different 3 random seeds in Table 10 (with different seed examples/training process), and observe that our method CLINGEN generally outperforms the baselines with non-negligible margins, which indicates the robustness of CLINGEN as it does not rely on a specific subset of few-shot training examples to perform well.

H Additional Quality Analysis

We present additional quality analysis of the synthetic dataset with t-SNE plots in Figure 12 and the regularized entity frequencies in Figure 13.

I Comparison with different prompt designs

I.1 Model Performance

We carry out an additional analysis with two recent and representative prompt optimization techniques, namely Reframe (Mishra et al., 2022), APE (Zhou et al., 2023) and PromptAgent (Wang et al., 2024).

In our setting, Reframe incorporates several principles (e.g. using low-level patterns, itemizing instructions, etc.) to produce high-quality prompts to enhance text generation, whereas APE and PromptAgent leverage the LLM to optimize the prompts based on the target task information. We demonstrate their performance on various clinical tasks in Table 11. The results indicate that our proposed CLINGEN consistently outperforms both baselines. This performance gain is attributed to the fact that the prompts generated by these baselines do not adequately address the unique challenges for the clinical data generation, i.e. distribution shift and lack of diversity. As a result, although they tend to include some generic task-specific information for guiding LLMs to generate training data, the performance gains brought by these advanced techniques are limited. One important avenue of future work is to design effective approach to combine these automatic prompt optimization approaches with our extracted clinical-related concepts.

I.2 Prompt Templates

We provide the detailed prompt templates we use for Reframe (Mishra et al., 2022), APE (Zhou et al., 2023) and PromptAgent (Wang et al., 2024) in the followings.

Natural Language Inference tasks:

Listing 14: Prompt Format for generating sentences in NLI tasks with Reframe.

Generate a pair of sentences for the [domain] task. Follow these guidelines:
1. Formulate a medical premise in the first sentence, such as a clinical observation or a patient's medical history.
2. Craft a medical hypothesis or claim related to the premise in the second sentence.
3. Ensure that the hypothesis logically follows from the premise.
4. Avoid introducing any unrelated or contradictory information in either sentence.
5. The length should be in 50 words.

Listing 15: Prompt Format for generating sentences in NLI tasks with APE.

Generate a pair of sentences for the [domain] task. The first sentence should be a medical premise, such as a clinical observation or a patient's medical history. The second sentence should be a medical hypothesis or claim, related to the premise. The goal is to determine whether the hypothesis logically follows from the premise, and you can use various medical scenarios, conditions, or treatments for creating these sentence pairs.
<table>
<thead>
<tr>
<th></th>
<th>LitCovid</th>
<th>CDR</th>
<th>MEDIQA-RQE</th>
<th>MQP</th>
<th>CHEMDNER</th>
<th>BC5CDR-Disease</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PubMedBERT</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reframe (Mishra et al., 2022)</td>
<td>56.74</td>
<td>57.27</td>
<td>61.92</td>
<td>67.60</td>
<td>54.61</td>
<td>59.17</td>
<td>59.55</td>
</tr>
<tr>
<td>APE (Zhou et al., 2023)</td>
<td>56.24</td>
<td>61.12</td>
<td>66.55</td>
<td>68.00</td>
<td>52.10</td>
<td>58.79</td>
<td>60.47</td>
</tr>
<tr>
<td>PromptAgent (Wang et al., 2024)</td>
<td>56.62</td>
<td>48.44</td>
<td>63.64</td>
<td>61.00</td>
<td>54.47</td>
<td>59.98</td>
<td>57.36</td>
</tr>
<tr>
<td>CLINGEN w/ KG</td>
<td>58.01</td>
<td>61.75</td>
<td>74.85</td>
<td>72.20</td>
<td>56.94</td>
<td>60.75</td>
<td>64.08</td>
</tr>
<tr>
<td>CLINGEN w/ LLM</td>
<td><strong>59.22</strong></td>
<td><strong>63.34</strong></td>
<td><strong>72.40</strong></td>
<td><strong>73.30</strong></td>
<td><strong>54.84</strong></td>
<td><strong>61.03</strong></td>
<td><strong>64.02</strong></td>
</tr>
</tbody>
</table>

| **PubMedBERT-large** |          |     |              |     |          |                |         |
| Reframe (Mishra et al., 2022) | 54.06 | 58.78 | 66.57 | 71.30 | 55.05 | 60.41 | 61.03 |
| APE (Zhou et al., 2023) | 53.54 | 61.65 | 69.20 | 71.00 | 53.03 | 59.87 | 61.38 |
| PromptAgent (Wang et al., 2024) | 54.54 | 50.10 | 65.56 | 64.20 | **55.91** | 62.17 | 58.75 |
| CLINGEN w/ KG | 55.81 | 62.66 | 79.92 | 75.40 | 55.56 | 61.21 | 65.16 |
| CLINGEN w/ LLM | **57.07** | **64.99** | **77.36** | **76.20** | **55.37** | **63.15** | **65.69** |

Table 11: Comparison between existing prompting optimization methods and CLINGEN.

Listing 16: Prompt Format for generating sentences in NLI tasks with PromptAgent.

You’ve been assigned the task of creating a dataset for determining the [domain] in medical text pairs. Ensure that you do not include any irrelevant information. Keep in mind that the content may involve medical conditions, treatments, and observations in various formats. Your goal is to accurately label the relationships for each medical text pair based on their logical connections.

**[domain]**: “Question Entailment” for MEDIQA-RQE.

**Sentence similarity tasks:**

Listing 17: Prompt Format for generating sentences in sentence similarity tasks with Reframe.

Suppose you need to generate two sentences for the [domain] task. The goal is to assess how close or similar the meaning of two sentences is, including 'equivalent' or 'not equivalent'.

Listing 18: Prompt Format for generating sentences in sentence similarity tasks with APE.

Suppose you need to generate two sentences for the [domain] task. The goal is to assess how close or similar the meaning of two sentences is, including 'equivalent' or 'not equivalent'.

Listing 19: Prompt Format for generating sentences in sentence similarity tasks with PromptAgent.

You’ve been assigned the job of creating a dataset for [domain]. Make sure not to include any extraneous details. Keep in mind that sentences can vary in structure and wording while conveying similar meanings. Your task is to calculate the similarity score accurately for each sentence pair.

**[domain]**: “Sentence Similarity Calculation” for MQP.

**Text classification tasks:**

Listing 20: Prompt Format for generating sentences in text classification tasks with Reframe.

Suppose you are a writer for [domain]. Your task is to give a synthetic [domain] about
[class_name] with the following instructions:
(1) Illustrate points with everyday scenarios related to the [class_name].
(2) about 50 - 100 words.

Listing 21: Prompt Format for generating sentences in text classification tasks with APE.

Suppose you are a writer for [domain]. Generate a clinical article discussing the latest advancements in [domain] with a focus on [class_name]. Please include information on recent clinical trials, emerging research findings, and potential implications for healthcare practitioners and patients.


You’ve been assigned the responsibility of creating a dataset for classifying text related to [domain]. Ensure that you do not include any irrelevant information. Keep in mind that references to COVID-19 may appear in various forms, including abbreviations and synonyms. Your objective is to accurately identify and classify text that is relevant to [domain].

[domain]: “COVID-19 Literature” for LitCovid.
[class_name]: the label name for this generated sample.

Relation extraction tasks:

Listing 23: Prompt Format for generating sentences in relation extraction tasks with Reframe.

Suppose you need to generate a dataset for the biomedical [domain] task where the relationships between entities in biomedical texts need to be identified. Your task is to give a synthetic example about [class_name] relation with the following instructions:

Listing 24: Prompt Format for relation extraction tasks with APE.

Generate a sentence that describes a [class_name] [domain] between [entity0] and [entity1]. The sentence should provide information about how these terms are related, such as its potential therapeutic use, side effects, or any relevant research findings.

Listing 25: Prompt Format for relation extraction tasks with PromptAgent.

You’ve been assigned the task of creating a [class_name] [domain] dataset for identifying relationships between [entity0] and [entity1] from the provided text. Be sure to exclude any extraneous information. Keep in mind that chemicals and diseases may be referred to using various names, abbreviations, or synonyms. Your goal is to recognize and extract these associations accurately.

[domain]: “Chemical Disease Relation” for CDR.
[entity0] and [entity1]: “chemical” and “disease: for CDR.
[class_name]: the label name for this generated sample.

Named entity recognition tasks:

Listing 26: Prompt Format for generating sentences in NER tasks with Reframe.

Suppose you need to create a dataset for [domain] recognition. Your task is to generate a sentence about [domain] and also output the [domain] name with the following instructions:

Listing 27: Prompt Format for generating sentences in NER tasks with APE.

(1) Provide the sentence or text snippet where the relationship is mentioned.
(2) The length should be in 50 words.

Listing 28: Prompt Format for generating sentences in NER tasks with PromptAgent.

You’ve been assigned the task of recognizing the [domain] name in the provided sentence. Be sure to exclude any extraneous information. Keep in mind that chemicals and diseases may be referred to using various names, abbreviations, or synonyms. Your goal is to recognize and extract these associations accurately.
recognizable entity type within the sentence.
(2) The named entity must be contextually relevant and correctly labeled with its type.
(3) The length should be in 50 words.

Listing 27: Prompt Format for NER tasks with APE.

Suppose you need to create a dataset for [domain] recognition. Generate a sentence or short text passage where you mention a [domain] entity within a context. The named entity should be clearly identifiable within the text.

Listing 28: Prompt Format for NER tasks with PromptAgent.

You’re tasked with generating a dataset for recognizing [domain] from the given sentence. Remember to avoid incorporating any associated elements. Consider both specific diseases and broader categories, and remember diseases and conditions can also appear as common abbreviations or variations.

[domain]: “disease” for BC5CDR-Disease; “chemical” for CHEMDNER.

J Using Medical LLMs as Data Generator

In this work, we mainly evaluate CLINGEN using GPT-family models as the LLM. However, we are aware that many LLMs have been fine-tuned on additional clinical contexts as well as instructions and achieved superior performance on clinical NLP benchmarks. We select MedAlpaca-13b (Han et al., 2023) as one representative clinical LLM and study the effect of CLINGEN using a medical LLM as the data generator. Many other medical LLMs, such as Med-PALM, are not open-sourced, thus we cannot run them in our experiments.

From the results shown in Table 12, we observe that using medical LLM as the clinical text data generator exhibits lower downstream performance. This could be attributed to the medical LLMs having fewer parameters than ChatGPT, which results in limited instruction-following capabilities.

<table>
<thead>
<tr>
<th>KG: LLM</th>
<th>LitCovid</th>
<th>CHEMDNER</th>
</tr>
</thead>
<tbody>
<tr>
<td>PubMedBERTbase</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLINGEN w/ KG</td>
<td>58.01</td>
<td>56.94</td>
</tr>
<tr>
<td>CLINGEN w/ LLM (ChatGPT)</td>
<td>59.22</td>
<td>54.84</td>
</tr>
<tr>
<td>CLINGEN w/ LLM (MedAlpaca)</td>
<td>55.45</td>
<td>52.15</td>
</tr>
<tr>
<td>PubMedBERTLarge</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLINGEN w/ KG</td>
<td>55.81</td>
<td>55.56</td>
</tr>
<tr>
<td>CLINGEN w/ LLM (ChatGPT)</td>
<td>57.07</td>
<td>55.37</td>
</tr>
<tr>
<td>CLINGEN w/ LLM (MedAlpaca)</td>
<td>53.90</td>
<td>52.67</td>
</tr>
</tbody>
</table>

Table 12: The performance of CLINGEN with the medical LLM MedAlpaca as data generator.

K Effect of Data Mixing Ratio

In this work, we present KGs and LLMs as two alternative and complementary sources for obtaining topics. However, we also consider combining topics from KGs and LLMs as a potential approach to enhance performance. Thus, we conduct experiments to demonstrate the impact of combining topics from KGs and LLMs at various ratios. Note that we still keep a total of 5000 generated synthetic samples to maintain a fair comparison. The experimental results in Table 13 indicate that combining knowledge from KGs and LLMs can yield a performance improvement, though not a substantial one. However, note that in practice, it is challenging to tune the ratio in the few-shot setting due to the limited volume of validation labels (Perez et al., 2021), and thus we only include the 1:1 results in Tables 7, 8, 9 in Appendix F for all the datasets.

<table>
<thead>
<tr>
<th>KG: LLM</th>
<th>LitCovid</th>
<th>CDR</th>
<th>MEDRQA-RQE</th>
<th>BC5CDR-Disease</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>PubMedBERTbase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>58.01</td>
<td>61.75</td>
<td>74.85</td>
<td>60.75</td>
<td>63.84</td>
</tr>
<tr>
<td>2.1</td>
<td>56.18</td>
<td>62.89</td>
<td>73.50</td>
<td>60.53</td>
<td>63.28</td>
</tr>
<tr>
<td>1.1</td>
<td>56.76</td>
<td>63.86</td>
<td>74.01</td>
<td>63.26</td>
<td>64.47</td>
</tr>
<tr>
<td>1.2</td>
<td>55.49</td>
<td>64.13</td>
<td>75.10</td>
<td>61.62</td>
<td>64.14</td>
</tr>
<tr>
<td>0.1</td>
<td>59.22</td>
<td>63.34</td>
<td>72.40</td>
<td>61.03</td>
<td>64.00</td>
</tr>
<tr>
<td>PubMedBERTLarge</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>55.81</td>
<td>62.66</td>
<td>79.92</td>
<td>61.21</td>
<td>64.90</td>
</tr>
<tr>
<td>2.1</td>
<td>54.21</td>
<td>64.22</td>
<td>76.15</td>
<td>62.40</td>
<td>64.25</td>
</tr>
<tr>
<td>1.1</td>
<td>56.80</td>
<td>65.90</td>
<td>79.12</td>
<td>65.94</td>
<td>66.94</td>
</tr>
<tr>
<td>1.2</td>
<td>54.41</td>
<td>64.68</td>
<td>80.77</td>
<td>64.55</td>
<td>66.10</td>
</tr>
<tr>
<td>0.1</td>
<td>57.07</td>
<td>64.99</td>
<td>77.36</td>
<td>63.15</td>
<td>65.64</td>
</tr>
</tbody>
</table>

Table 13: Effect of mixing topics generated from KG and LLM in different ratio.