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 gemeral-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 entity 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$_{\text{Base}}$ and 7.7% for PubMedBERT$_{\text{Large}}$).

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.

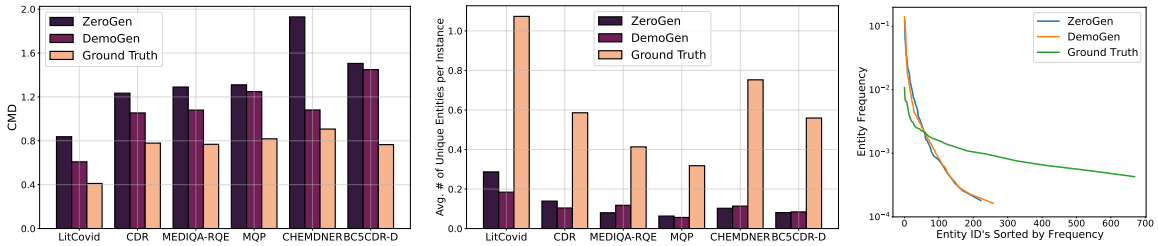
On the other hand, several works aimed at optimizing prompts using LLMs (Zhou et al., 2023; Wang et al., 2024) or knowledge graphs (Liu et al., 2022b; Chen et al., 2022b), yet they mainly focus on refining prompts to obtain the answer for the given input, and the prompt template often remains unchanged. 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 $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{K}$, where (x_i, y_i) represents an input text and its corresponding label



(a) CMD (b) Entity Coverage (c) Entity Frequency

Figure 1: Preliminary Studies. (c) is from BC5CDR-Disease and is in log scale.

$y_i \in \mathcal{Y}$ for the i-th example. K denotes the total number of training samples, which is kept at a very small value (5-shot per label). The primary objective is to harness the LLM $\mathcal{M}$ to generate a synthetic dataset, denoted as $\widetilde{\mathcal{D}} = \{(\widetilde{x}_i, \widetilde{y}_i)\}_{i=1}^{N}$, where N is the number of generated samples ($N \gg K$). We use $\rho(\cdot)$ to denote the generation process from the LLM. For each downstream task, we fine-tune a classifier $\mathcal{C}_\theta$ (a moderate-size pre-trained language model) parameterized by θ on the synthetic dataset $\widetilde{\mathcal{D}}$ for evaluating its quality.[1]

3.2 Limitations of Existing Methods

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 as $\widetilde{\mathcal{D}}_{\text{Zero}} \sim \rho_{j\sim\mathcal{Y}}(\cdot; p_j)$, and DemoGen (Yoo et al., 2021; Meng et al., 2023), which augments the prompt with few-shot demonstrations $\mathcal{D}$ as $\widetilde{\mathcal{D}}_{\text{Demo}} \sim \rho_{j\sim\mathcal{Y}}(\cdot; [p_j, \mathcal{D}])$. 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

[1] While In-context Learning (Brown et al., 2020) can also be utilized, it is often hard to fit all generated instances into the context window, especially for datasets with high cardinality.

[2] Details of calculating CMD is in Appendix A.

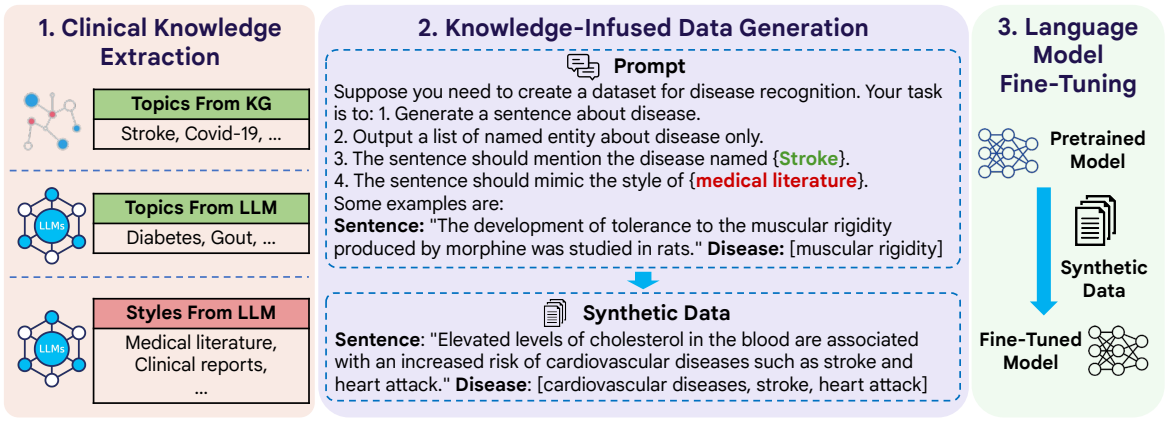


Figure 2: The overview of CLINGEN.

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 $\mathcal{M}$. 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 (iBKH) as the KG (Su et al., 2023) $\mathcal{G}$ to generate topics $\mathcal{T}_{\text{KG}} \sim \text{query}(\mathcal{G})$ 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 iBKH to bolster the medical information as $\mathcal{T}_{\text{KG}}^{\text{NCBI}} \sim \text{query}(\mathcal{G}_{\text{disease}})$, $\mathcal{G}_{\text{disease}} = \{e \in \mathcal{G} | \text{type}(e) = \text{disease}\}$. 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 $\mathcal{T}_{\text{KG}}^{\text{CDR}} \sim \text{query}(\mathcal{G}_{\text{relation\_cd}})$, $\mathcal{G}_{\text{relation\_cd}} = \{\langle c, r, d\rangle \in \mathcal{G} | \text{type}(r) = \text{has\_relation}\}$. 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.*, "*Suppose you are a clinician and want to collect a set of <Entity Type>. 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^{style} (e.g., *disease entity recognition*, *recognizing text entailment*) and integrate few-shot

demonstrations d_i^{style}. We then engage LLM in suggesting several potential sources, speakers, or authors of the sentences as $\mathcal{W} \sim \rho\left(\cdot; [p_i^{\text{style}}, d_i^{\text{style}}]\right)$. Responses such as "*medical literature*" or "*patient-doctor dialogues*" are augmented into the prompts to imitate the writing styles found in real datasets.

4.2 Knowledge-infused Data Generation

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 $\widetilde{\mathcal{D}}_{\text{Clin}}$ as

$$\widetilde{\mathcal{D}}_{\text{Clin}} \sim \rho_{j\sim\mathcal{Y},t\sim\mathcal{T},w\sim\mathcal{W}}\left(\cdot; [p_j, t, w], \mathcal{D}\right).$$

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

$$\theta^{(1)} = \min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}} \ell\left(f(x;\theta), y\right),$$
$$\theta^{(2)} = \min_\theta \mathbb{E}_{(\widetilde{x},\widetilde{y})\sim\widetilde{\mathcal{D}}} \ell\left(f(\widetilde{x};\theta), \widetilde{y}\right), \theta_{\text{init}} = \theta^{(1)}.$$

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 $\mathcal{M}$[3] 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

[3]Studies on using Medical LLMs are in Appendix J.

Task	*Single-Sentence Tasks*		*Sentence-Pair Tasks*					*Token Classification Tasks*				
	Text Class (2)	**RE (3)**	**NLI (3)**	**Fact Verification (2)**		**STS (1)**	**QA (2)**	**NER (4)**		**MedAttr (1)**		
	F1	F1	Acc	Acc	F1	Acc	Acc	F1	F1-subset*	P	R	F1
PubMedBERT$_{\texttt{Base}}$												
Supervised-Full	77.01	77.34	79.20	67.58	65.49	75.70	74.70	89.67	87.27	—	—	—
Supervised-Few	18.61	43.89	44.64	29.43	27.10	55.70	54.74	39.41	34.12	38.11	43.82	40.77
DA-Word Sub (2020)	40.74	38.14	55.08	28.86	25.83	54.40	53.58	44.30	40.41	40.25	47.65	43.64
DA-Back Trans (2020)	47.24	—	54.30	32.15	28.04	55.80	53.28	—	—	—	—	—
DA-Mixup (2020; 2020)	45.09	43.37	53.52	32.78	29.12	58.20	51.91	42.20	37.65	42.37	48.96	45.43
DA-Transformer (2022; 2020)	41.02	47.56	55.71	35.32	31.77	58.80	56.36	44.75	39.66	37.82	44.28	40.80
LightNER† (2022a)	—	—	—	—	—	—	—	—	39.49	—	—	—
KGPC† (2023)	—	—	—	—	—	—	—	—	51.60	—	—	—
ZeroGen (2022a; 2022)	59.02	63.84	55.96	35.30	32.50	68.35	61.89	56.97	48.26	52.80	49.53	51.11
DemoGen (2023; 2021)	64.09	67.46	59.80	40.30	35.95	70.85	62.01	60.16	53.91	58.15	56.84	57.49
ProGen (2022b)	65.16	67.23	59.57	37.71	34.54	69.30	60.74	60.49	55.11	57.76	58.57	58.16
S3 (2023)	65.12	67.60	61.36	40.17	36.44	70.20	63.58	60.36	54.25	56.21	63.60	59.68
CLINGEN w/ KG	67.15	69.01	64.89	43.83	39.43	72.20	**71.49**	**64.26**	**60.11**	**71.75**	65.20	**68.32**
CLINGEN w/ LLM	**67.82**	**70.06**	**67.24**	**46.50**	**41.46**	**73.30**	69.60	63.17	58.49	68.19	**66.79**	67.48
Performance Gain	4.08%	3.63%	9.58%	15.38%	13.77%	3.47%	12.44%	6.23%	—	—	—	14.48%
PubMedBERT$_{\texttt{Large}}$												
Supervised-Full	80.06	79.64	82.65	72.97	69.23	78.80	80.37	90.15	87.68	—	—	—
Supervised-Few	17.86	52.68	50.00	40.90	30.50	59.73	59.50	42.84	37.57	41.30	45.02	43.08
DA-Word Sub (2020)	43.99	44.35	57.66	35.51	31.95	55.30	58.57	46.67	43.70	46.77	43.52	45.09
DA-Back Trans (2020)	50.98	—	58.39	34.12	31.36	56.40	57.19	—	—	—	—	—
DA-Mixup (2020; 2020)	46.74	50.97	57.35	34.01	31.10	58.50	56.68	46.69	43.01	41.25	52.09	46.04
DA-Transformer (2022; 2020)	44.41	46.12	58.94	35.09	30.95	58.10	59.30	46.94	43.50	43.36	45.78	44.54
ZeroGen (2022a; 2022)	61.51	65.18	63.47	41.12	36.10	72.69	66.02	57.79	49.10	54.04	51.40	52.69
DemoGen (2023; 2021)	64.97	68.65	64.58	42.61	38.69	74.37	65.04	61.43	55.61	62.67	61.02	61.83
ProGen (2022b)	65.01	69.23	63.32	42.79	38.63	74.90	63.27	62.47	57.31	57.21	63.70	60.28
S3 (2023)	64.33	69.65	65.07	41.76	37.72	73.20	66.33	61.97	56.29	63.07	62.72	62.89
CLINGEN w/ KG	66.76	71.47	**70.90**	48.62	42.45	75.40	**73.94**	**65.48**	**62.23**	70.96	**69.66**	**70.30**
CLINGEN w/ LLM	**67.61**	**72.81**	70.50	**49.51**	**43.72**	**76.21**	73.40	65.36	61.89	**71.61**	66.86	69.15
Performance Gain	4.00%	4.54%	8.96%	15.70%	13.00%	3.47%	11.47%	1.76%	—	—	—	11.78%

Table 1: Experimental results aggregated by tasks. **Bold** and underline denote the best and second-best results. †: Models exclusive to NER tasks. ∗: Since the two † models only report results on two NER datasets, we report the average performance on those two datasets for a fair comparison. "Supervised-Full" and "Supervised-Few" denote the results using the original dataset and using only the few-shot examples as training data, respectively.

4 LLM-based data generation techniques. See Appendix D for their descriptions.

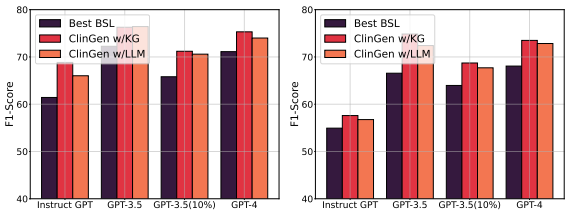
Implementation Details. For implementation, we use PyTorch (Paszke et al., 2019) and HuggingFace (Wolf et al., 2019). For each dataset, we randomly sample 5 examples from each class to provide few-shot demonstrations and keep a validation set of the same size. During the data generation process when we call the ChatGPT APIs (OpenAI, 2023b), we set the parameter 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.

[4]We do not further increase t, as previous analysis (Chung et al., 2023; Yu et al., 2023) has shown that increasing t to larger value does not help with additional performance gain.

5.2 Model Performance with Synthetic Data

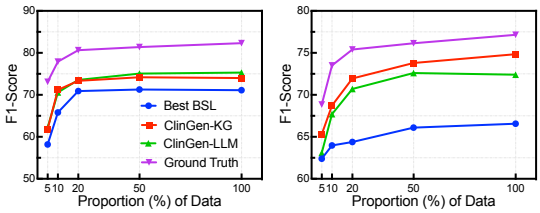
Table 1 summarizes the experimental results. Due to space limits, we report the average performance over all datasets for each task, but provide the detailed results for each dataset in Tables 7, 8, 9 in Appendix F. Based on the experimental results, we have the following findings:

◇ 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,



(a) HOC (b) MEDIQA-RQE

Figure 3: Different generators at Base.



(a) HOC (b) MEDIQA-RQE

Figure 4: Different proportion of data at Base.

	HOC	GAD			ChemProt	MEDIQA-RQE	PUBHEALTH		NCBI-Disease			CASI		
	F1	P	R	F1	F1	ACC	ACC	F1	P	R	F1	P	R	F1
ChatGPT Inference (OpenAI)	68.76	84.21	**97.46**	90.35	49.42	74.31	**69.50**	**52.47**	46.62	52.31	49.30	48.82	**74.75**	59.07
PMC-LLaMa-13B Inference (Wu et al.)	50.07	89.61	81.18	85.19	33.35	52.17	48.01	32.84	27.11	23.97	25.44	56.38	36.87	41.58
MedAlpaca-13B Inference (Han et al.)	40.44	71.95	72.48	72.21	31.29	58.12	55.40	34.63	44.69	31.16	27.85	52.51	49.16	51.64
CLINGEN w/ KG	77.71	94.30	89.09	**91.62**	60.12	**79.92**	50.20	41.26	**62.46**	**64.08**	**63.26**	70.96	69.66	**70.30**
CLINGEN w/ LLM	**78.14**	**95.08**	86.14	90.39	**63.05**	77.36	52.96	43.31	61.12	60.16	60.64	**71.61**	66.86	69.15

Table 2: Comparison between prompting LLM for inference and CLINGEN at Large scale.

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

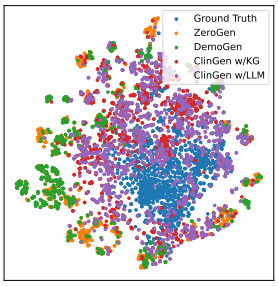
Effect of Different LLM Generators. To investigate the impact of various LLMs on CLINGEN, we utilize InstructGPT (`text-curie-001`) (Ouyang et al., 2022) and GPT-4 (OpenAI, 2023a). Note that we only generate 500 samples in the GPT-4 setting due to budget constraints, but we provide the results of GPT-3.5 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.

	HOC		CDR		MEDIQA-RQE		NCBI-Disease	
	w/ KG	w/ LLM	w/ KG	w/ LLM	w/ KG	w/ LLM	w/ KG	w/ LLM
CLINGEN	**76.28**	**76.42**	**61.74**	**63.34**	**74.85**	**72.40**	**59.46**	**55.95**
w/o Styles	73.25	74.40	59.10	60.15	67.21	66.50	57.97	54.70
w/o Topics	70.86		58.51		69.86		55.09	

Table 3: Ablation studies on topic extraction and style suggestion at Base scale.

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 PubMedBERT$_{\texttt{Large}}$ 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 PubMedBERT$_{\texttt{Large}}$ 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



(a) t-SNE plot

Entail

Sentence A: I've been experiencing a discomfort in my stomach, what could be causing it?
Sentence B: What are the possible causes for abdominal pain?

Sentence A: I recently started working with metal and found out about the health risks of beryllium exposure. What are the symptoms of beryllium poisoning, and how can I protect myself from it?
Sentence B: What are the symptoms and preventive measures for berylliosis?

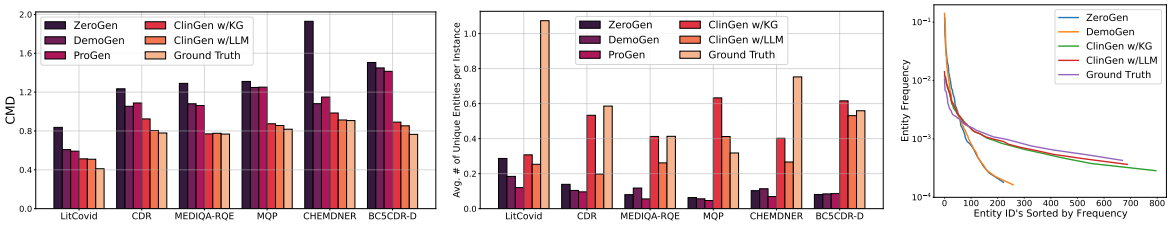
Not Entail

Sentence A: Why are my nails turning yellow? It's never happened before.
Sentence B: What are some home remedies for acne scars?

Sentence A: I feel like food is getting stuck in my throat, is there anything I can do to make it easier to swallow?
Sentence B: What are some home remedies for a sore throat?

(b) Case study of generated examples

Figure 5: Data distribution and diversity measures on CLINGEN. (a) is from BC5CDR-Disease and (b) is from MEDIQA-RQE using CLINGEN with LLM.



(a) CMD (b) Entity Coverage (c) Entity Frequency

Figure 6: Data distribution and diversity measures on CLINGEN. (c) is from BC5CDR-Disease.

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.

	HOC	CDR	MEDIQA-RQE	NCBI-Disease
ZeroGen	0.512	0.469	0.277	0.528
DemoGen	0.463	0.377	0.289	0.281
ProGen	0.481	0.321	0.290	0.357
CLINGEN w/ KG	0.440	**0.291**	**0.243**	0.180
CLINGEN w/ LLM	**0.432**	0.338	0.255	**0.155**
Ground truth	0.265	0.268	0.164	0.262

Table 4: Average Pairwise Similarity.

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

	HOC	GAD	ChemProt	MEDIQA-RQE	PUBHEALTH	NCBI-Disease	CASI
GPT-3.5 Inference	1.09	1.05	5.75	2.15	2.80	0.90	1.30
DemoGen	0.59	0.66	1.35	0.81	0.92	1.12	1.28
CLINGEN w/ KG	0.65	0.73	1.47	0.86	1.01	1.41	1.55
CLINGEN w/ LLM	0.72	0.84	1.51	0.90	1.34	1.49	1.62

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.

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, Mediqa-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.

7 Conclusion

In this work, 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 Form[5]. This allows us to use the Azure OpenAI service while ensuring Microsoft does not have access to patient data.

References

Asma Ben Abacha and Dina Demner-Fushman. 2016. Recognizing question entailment for medical question answering. In *AMIA Annual Symposium Proceedings*, volume 2016, page 310.

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1998–2022, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. 2015. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440.

Asma Ben Abacha, Chaitanya Shivade, and Dina Demner-Fushman. 2019. Overview of the MEDIQA 2019 shared task on textual inference, question entailment and question answering. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 370–379, Florence, Italy. Association for Computational Linguistics.

Àlex 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).

[5] https://aka.ms/oai/additionalusecase

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Peng Chen, Jian Wang, Hongfei Lin, Di Zhao, and Zhihao Yang. 2023. Few-shot biomedical named entity recognition via knowledge-guided instance generation and prompt contrastive learning. *Bioinformatics*, 39(8):btad496.

Qingyu Chen, Alexis Allot, Robert Leaman, Rezarta Islamaj Doğan, and Zhiyong Lu. 2021. Overview of the biocreative vii litcovid track: multi-label topic classification for covid-19 literature annotation. In *Proceedings of the BioCreative challenge evaluation workshop*.

Xiang Chen, Lei Li, Shumin Deng, Chuanqi Tan, Changliang Xu, Fei Huang, Luo Si, Huajun Chen, and Ningyu Zhang. 2022a. LightNER: A lightweight tuning paradigm for low-resource NER via pluggable prompting. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2374–2387, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022b. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM Web conference 2022*, pages 2778–2788.

Bharath Chintagunta, Namit Katariya, Xavier Amatriain, and Anitha Kannan. 2021. Medically aware GPT-3 as a data generator for medical dialogue summarization. In *Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations*, pages 66–76, Online. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *ArXiv preprint*, abs/2210.11416.

John Chung, Ece Kamar, and Saleema Amershi. 2023. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 575–593,

Toronto, Canada. Association for Computational Linguistics.

Rezarta Islamaj Dogan, 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.

Jason Alan Fries, Leon Weber, Natasha Seelam, Gabriel Altay, Debajyoti Datta, Samuele Garda, Myungsun Kang, Ruisi Su, Wojciech Kusa, Samuel Cahyawijaya, Fabio Barth, Simon Ott, et al. 2022. Bigbio: A framework for data-centric biomedical natural language processing. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

John Giorgi, Augustin Toma, Ronald Xie, Sondra Chen, Kevin R An, Grace X Zheng, and Bo Wang. 2023. Clinical note generation from doctor-patient conversations using large language models: Insights from mediqa-chat. *ArXiv preprint*, abs/2305.02220.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. 2023. Medalpaca–an open-source collection of medical conversational ai models and training data. *ArXiv preprint*, abs/2304.08247.

Julia Ive, Natalia Viani, Joyce Kam, Lucia Yin, Somain Verma, Stephen Puntis, Rudolf N Cardinal, Angus Roberts, Robert Stewart, and Sumithra Velupillai. 2020. Generation and evaluation of artificial mental health records for natural language processing. *NPJ digital medicine*, 3(1):69.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. 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 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Tian Kang, Adler Perotte, Youlan Tang, Casey Ta, and Chunhua Weng. 2021. Umls-based data augmentation for natural language processing of clinical research literature. *Journal of the American Medical Informatics Association*, 28(4):812–823.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 5189–5197. AAAI Press.

Neema Kotonya and Francesca Toni. 2020. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7740–7754, Online. Association for Computational Linguistics.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M. Lowe, Roger A. Sayle, Riza Batista-Navarro, et al. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of Cheminformatics*, 7(1):S2.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. In *Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems*, pages 18–26, Suzhou, China. Association for Computational Linguistics.

Peter Lee, Carey Goldberg, and Isaac Kohane. 2023. *The AI Revolution in Medicine: GPT-4 and Beyond*. Pearson Education, Limited.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. Biocreative V CDR task corpus: a resource for chemical disease relation extraction. *Database J. Biol. Databases Curation*, 2016.

Michelle M Li, Kexin Huang, and Marinka Zitnik. 2022. Graph representation learning in biomedicine and healthcare. *Nature Biomedical Engineering*, 6(12):1353–1369.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023. Synthetic data generation with large language models for text classification: Potential and limitations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10443–10461, Singapore. Association for Computational Linguistics.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022a. WANLI: Worker and AI collaboration for natural language inference dataset creation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6826–6847, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022b. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.

Jialin Liu, Changyu Wang, and Siru Liu. 2023. Utility of chatgpt in clinical practice. *Journal of Medical Internet Research*, 25:e48568.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Clara H. McCreery, Namit Katariya, Anitha Kannan, Manish Chablani, and Xavier Amatriain. 2020. Effective transfer learning for identifying similar questions: Matching user questions to COVID-19 faqs. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3458–3465. ACM.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. 2022. Generating training data with language models: Towards zero-shot language understanding. In *Advances in Neural Information Processing Systems*.

Yu Meng, Martin Michalski, Jiaxin Huang, Yu Zhang, Tarek Abdelzaher, and Jiawei Han. 2023. Tuning language models as training data generators for augmentation-enhanced few-shot learning. In *International Conference on Machine Learning*, pages 24457–24477. PMLR.

Bertalan Meskó and Eric J Topol. 2023. The imperative for regulatory oversight of large language models (or generative ai) in healthcare. *NPJ Digital Medicine*, 6(1):120.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2022. Reframing instructional prompts to GPTk's language. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 589–612, Dublin, Ireland. Association for Computational Linguistics.

Sungrim Moon, Serguei Pakhomov, Nathan Liu, James O Ryan, and Genevieve B Melton. 2014. A sense inventory for clinical abbreviations and acronyms created using clinical notes and medical dictionary resources. *Journal of the American Medical Informatics Association*, 21(2):299–307.

OpenAI. 2023a. Gpt-4 technical report. *arXiv*.

OpenAI. 2023b. Introducing chatgpt.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*, pages 8024–8035.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 11054–11070.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Mourad Sarrouti, Asma Ben Abacha, Yassine Mrabet, and Dina Demner-Fushman. 2021. Evidence-based fact-checking of health-related claims. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3499–3512, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chaitanya Shivade. 2017. Mednli — a natural language inference dataset for the clinical domain.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023. Large language models encode clinical knowledge. *Nature*.

Chang Su, Yu Hou, Manqi Zhou, Suraj Rajendran, Jacqueline RMA Maasch, Zehra Abedi, Haotan Zhang, Zilong Bai, Anthony Cuturrufo, Winston Guo, et al. 2023. Biomedical discovery through the integrative biomedical knowledge hub (ibkh). *Iscience*, 26(4).

Olivier Taboureau, Sonny Kim Nielsen, Karine Audouze, Nils Weinhold, Daniel Edsgärd, Francisco S Roque, Irene Kouskoumvekaki, Alina Bora, et al. 2010. Chemprot: a disease chemical biology database. *Nucleic acids research*, 39:D367–D372.

Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. 2023. Does synthetic data generation of llms help clinical text mining? *ArXiv preprint*, abs/2303.04360.

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

Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):1–28.

Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Chuck Lau, Ryutaro Tanno, Ira Ktena, et al. 2023. Towards generalist biomedical ai. *ArXiv preprint*, abs/2307.14334.

Ruida Wang, Wangchunshu Zhou, and Mrinmaya Sachan. 2023. Let's synthesize step by step: Iterative dataset synthesis with large language models by extrapolating errors from small models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P Xing, and Zhiting Hu. 2024. Promptagent: Strategic planning with language models enables expert-level prompt optimization. In *The Twelfth International Conference on Learning Representations*.

Zifeng Wang and Jimeng Sun. 2022. PromptEHR: Conditional electronic healthcare records generation with prompt learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2873–2885, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. 2016. Assessing the state of the art in biomedical relation extraction: overview of the biocreative v chemical-disease relation (cdr) task. *Database*, 2016.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv preprint*, abs/1910.03771.

Michael Wornow, Yizhe Xu, Rahul Thapa, Birju Patel, Ethan Steinberg, Scott Fleming, Michael A Pfeffer, Jason Fries, and Nigam H Shah. 2023. The shaky foundations of clinical foundation models: A survey of large language models and foundation models for emrs. *ArXiv preprint*, abs/2303.12961.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. Pmc-llama: Further finetuning llama on medical papers. *ArXiv preprint*, abs/2304.14454.

Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022a. ZeroGen: Efficient zero-shot learning via dataset generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11653–11669, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jiacheng Ye, Jiahui Gao, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2022b. ProGen: Progressive zero-shot dataset generation via in-context feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3671–3683, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kang Min Yoo, Dongju Park, Jaewook Kang, Sang-Woo Lee, and Woomyoung Park. 2021. GPT3Mix: Leveraging large-scale language models for text augmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2225–2239, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. 2023. Large language model as attributed training data generator: A tale of diversity and bias. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschläger, and Susanne Saminger-Platz. 2017. Central moment discrepancy (CMD) for domain-invariant representation learning. In *5th International Conference on Learning Representations*.

Rongzhi Zhang, Yue Yu, and Chao Zhang. 2020. SeqMix: Augmenting active sequence labeling via sequence mixup. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8566–8579, Online. Association for Computational Linguistics.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *ArXiv preprint*, abs/2309.01219.

Ran Zhou, Xin Li, Ruidan He, Lidong Bing, Erik Cambria, Luo Si, and Chunyan Miao. 2022. MELM: Data augmentation with masked entity language modeling for low-resource NER. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2251–2262, Dublin, Ireland. Association for Computational Linguistics.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2023. Large language models are human-level prompt engineers. In *The Eleventh International Conference on Learning Representations*.

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

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

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

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

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

$$\mathrm{CMD}_k(p, q) = \frac{1}{|b-a|}\|\mathbf{E}(X) - \mathbf{E}(Y)\|_2 + \sum_{k=2}^{K} \frac{1}{|b-a|^k} \|C_k(X) - C_k(Y)\|_2$$

where $\mathbf{E}(X) = \frac{1}{|X|}\sum_{x \in X} x$ is the empirical expectation vector computed on the sample X and $C_k(X) = \mathbf{E}\left((x - \mathbf{E}(X))^k\right)$ is the vector of all k^{th} order sample central moments of the coordinates of X[6]. To adapt CMD in our work, we set $K = 5$, and use the embedding from Sentence-BERT (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.

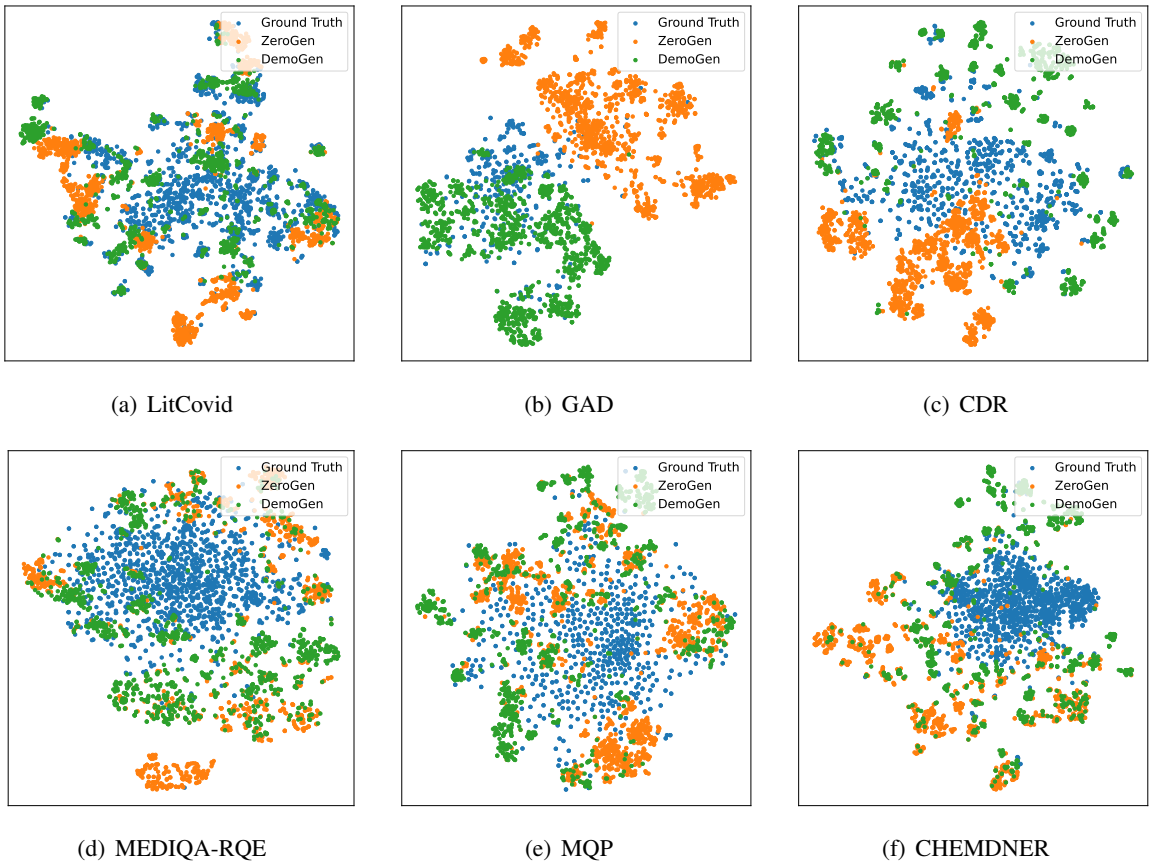
C Dataset Description

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/yusuke0519/724aa68fc431afadb0cc7280168da17b

[7] We employ SentenceBERT (Reimers and Gurevych, 2019) as the text encoder.



(a) LitCovid (b) GAD (c) CDR

(d) MEDIQA-RQE (e) MQP (f) CHEMDNER

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):

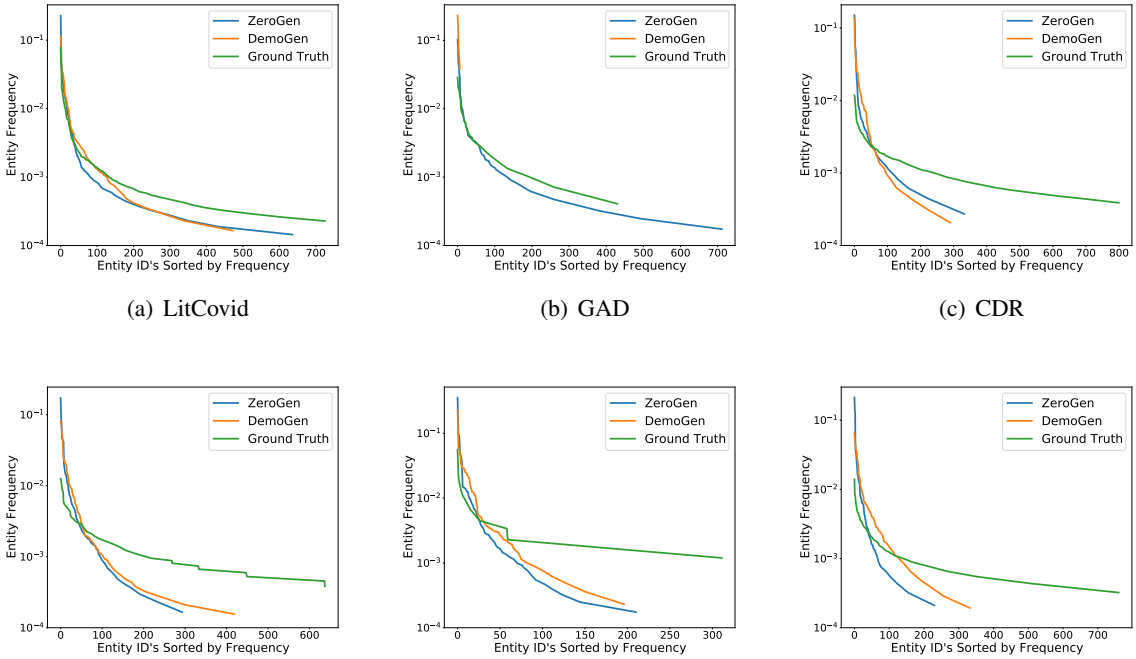
  - The *MedNLI* (Shivade, 2017) dateset consists of sentences pairs derived from MIMIC-III, where we predict the relations between the sentences. The labels include “entailment", “neutral" and “contradiction".
  - The *MEDIQA-NLI* (Ben Abacha et al., 2019) dataset comprises text-hypothesis pairs. Their 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-



(a) LitCovid (b) GAD (c) CDR

(d) MEDIQA-RQE (e) MQP (f) CHEMDNER

Figure 8: The regularized entity frequencies of datasets generated by ZeroGen and DemoGen compared with the ground truth in log scale.

	ZeroGen	DemoGen	Ground Truth
Entail	**Sentence A:** Can drinking alcohol increase the risk of liver disease? **Sentence B:** Does alcohol consumption contribute to liver disease risk?	**Sentence A:** What are the side effects of chemotherapy? **Sentence B:** What are the possible adverse effects of chemotherapy?	**Sentence A:** My 3yrs old boy found my bleach at the laundry and I suspect he swallowed a bit of it. How do I treat this pls. **Sentence B:** What the Doc will do if a child swallows bleach?
Not Entail	**Sentence A:** What are the side effects of metformin? **Sentence B:** Can I take ibuprofen for a headache?	**Sentence A:** What are the common symptoms of influenza? **Sentence B:** Can I take ibuprofen to manage my headache?	**Sentence A:** I have exercise induced asthma. Would any of these non drug devises be suitable please? **Sentence B:** Are there any treatments or cures for albinism?

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.

Corpus	Tasks	#Class	#Train/#Test	Metrics
Single-Sentence Tasks				
LitCovid (Chen et al., 2021)	Text Classification	7	24960/6238	**F1**
HOC (Baker et al., 2015)	Text Classification	10	3091/898	**F1**
GAD (Bravo et al., 2015)	Relation Extraction (RE)	1	4750/350	P, R, **F1**
CDR (Wei et al., 2016)	Relation Extraction (RE)	1	8431/2522	P, R, **F1**
ChemProt (Taboureau et al., 2010)	Relation Extraction (RE)	5	8793/1087	**F1**
Sentence-Pair Tasks				
MedNLI* (Shivade, 2017)	Natural Language Inference (NLI)	3	11232/1422	**Acc**
MEDIQA-NLI† (Ben Abacha et al., 2019)	Natural Language Inference (NLI)	3	-/405	**Acc**
MEDIQA-RQE (Abacha and Demner-Fushman, 2016)	Natural Language Inference (NLI)	2	8588/302	**Acc**
PUBHEALTH (Kotonya and Toni, 2020)	Fact Verification	4	9804/1231	Acc, **F1**
HealthVer (Sarrouti et al., 2021)	Fact Verification	3	10591/1824	Acc, **F1**
MQP (McCreery et al., 2020)	Sentences Similarity (STS)	2	10/3033	**Acc**
PubmedQA (Jin et al., 2019)	Question Answering (QA)	2	500/500	**Acc**
BioASQ (Tsatsaronis et al., 2015)	Question Answering (QA)	2	670/140	**Acc**
Token Classification Tasks				
BC5CDR-Disease (Li et al., 2016)	Named Entity Recognition (NER)	1	4882/5085	P, R, **F1**
BC5CDR-Chemical (Li et al., 2016)	Named Entity Recognition (NER)	1	4882/5085	P, R, **F1**
NCBI-Disease (Dogan et al., 2014)	Named Entity Recognition (NER)	1	5336/921	P, R, **F1**
CHEMDNER (Krallinger et al., 2015)	Named Entity Recognition (NER)	1	14522/12430	P, R, **F1**
CASI (Agrawal et al., 2022; Moon et al., 2014)	Attribute Extraction	6	5/100	**F1**

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 $\mathcal{D}$ 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].
```

[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 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].
```

[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 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].
```

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

Listing 7: Prompt Format for generating the second sentence in NLI tasks with CLINGEN.

```
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:
1. generate one short [content]
about [topic] so that [label_desc].
2. The [content] should mimic the
style of the first sentence.
```

[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

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.
```

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

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.

	LitCovid	HOC	CDR			GAD			ChemProt
	F1	F1	P	R	F1	P	R	F1	F1
PubMedBERT$_{\text{Base}}$									
Supervised-Full (SOTA)	73.55	84.35	67.81	76.60	71.96	—	—	84.39	77.97
Supervised-Full	71.70	82.32	67.81	76.60	71.96	82.55	85.10	83.81	76.24
Supervised-Few	24.08	13.13	41.62	52.96	46.61	57.71	46.54	51.53	33.54
DA-Word Sub	36.49	44.98	40.50	46.20	43.16	51.15	32.10	39.45	31.82
DA-Back Trans	39.70	54.78	—	—	—	—	—	—	—
DA-Mixup	40.82	49.35	41.40	44.80	43.03	55.44	48.30	51.62	35.45
DA-Transformer	39.86	42.18	44.60	61.70	51.77	59.40	46.50	52.16	38.73
ZeroGen	50.50	67.90	38.82	**91.82**	54.57	84.38	80.68	82.49	54.46
DemoGen	57.65	70.52	46.90	83.3	60.01	93.14	80.19	86.18	56.18
ProGen	58.06	72.25	51.35	71.58	59.80	90.52	**85.14**	87.75	54.15
S3	58.67	71.58	49.76	76.08	60.17	94.85	80.19	86.90	55.75
CLINGEN w/ KG	58.01	76.28	56.98	67.38	61.75	93.33	83.68	**88.24**	57.04
CLINGEN w/ LLM	**59.22**	**76.42**	**60.60**	66.35	**63.34**	**94.61**	78.17	85.61	**61.22**
CLINGEN w/ KG+LLM	56.56	78.02	57.97	71.09	63.86	92.57	88.59	90.54	58.48
PubMedBERT$_{\text{Large}}$									
Supervised-Full (SOTA)	—	84.87	—	—	—	—	—	84.90	78.77
Supervised-Full	74.59	85.53	72.31	74.88	73.57	84.95	88.75	86.81	78.55
Supervised-Few	22.59	13.13	42.27	67.51	51.99	57.58	90.07	70.25	35.80
DA-Word Sub	37.20	50.78	47.70	43.50	45.50	63.40	42.00	50.53	37.01
DA-Back Trans	40.50	61.46	—	—	—	—	—	—	—
DA-Mixup	40.03	53.45	43.34	73.50	54.53	62.20	59.93	60.52	37.87
DA-Transformer	38.95	49.86	50.70	31.60	38.93	59.80	57.76	58.76	40.66
ZeroGen	52.86	70.16	42.95	**80.67**	56.06	92.26	76.73	83.78	55.71
DemoGen	56.29	73.65	50.86	74.30	60.39	**96.85**	76.83	85.69	59.88
ProGen	54.71	75.31	50.36	76.08	60.60	91.11	85.63	88.29	58.79
S3	53.56	75.11	51.51	78.30	62.14	92.12	83.80	87.76	59.05
CLINGEN w/ KG	55.81	77.71	60.45	65.04	62.66	94.30	**89.08**	**91.62**	60.12
CLINGEN w/ LLM	**57.07**	**78.14**	**67.13**	62.98	**64.99**	95.08	86.14	90.39	**63.05**
CLINGEN w/ KG+LLM	56.80	79.07	64.19	67.70	65.90	92.41	92.07	92.24	59.95

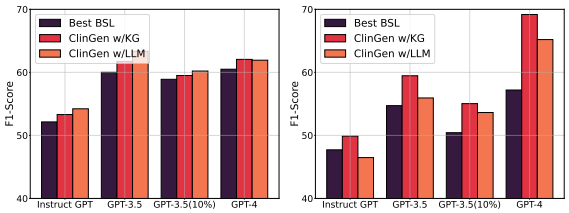
Table 7: Performance on single-sentence tasks evaluated by PubMedBERT$_{\text{Base}}$ and PubMedBERT$_{\text{Large}}$. **Bold** and underline indicate the best and second best results for each dataset, respectively. Note that the performance of 'Supervised-Full (SOTA)' is copied from the existing paper. 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).

	MEDIQA-RQE	MEDIQA-NLI	MedNLI	PUBHEALTH		HealthVer		MQP	PubmedQA	BioASQ
	ACC	ACC	ACC	ACC	F1	ACC	F1	ACC	ACC	ACC
PubMedBERT$_{\texttt{Base}}$										
Supervised-Full (SOTA)	—	—	86.60	70.52	69.73	73.54	74.82	79.20	70.20	91.43
Supervised-Full	77.15	79.01	81.43	65.16	62.96	70.00	68.02	75.70	61.84	87.56
Supervised-Few	57.51	40.00	36.40	28.30	23.70	30.55	30.49	55.70	55.90	53.57
DA-Word Sub	58.60	50.24	56.40	23.67	17.64	34.05	34.02	54.40	52.88	54.28
DA-Back Trans	59.16	49.92	53.82	30.70	23.32	33.60	32.76	55.80	53.70	52.86
DA-Mixup	57.71	49.38	53.47	31.45	24.45	34.11	33.78	58.20	51.68	52.14
DA-Transformer	62.25	51.19	53.70	34.81	27.75	35.83	35.78	58.80	54.14	58.57
ZeroGen	63.28	52.89	57.71	35.80	31.50	34.80	33.50	68.35	55.20	68.57
DemoGen	66.56	56.29	58.56	42.60	35.40	38.00	36.50	70.85	57.60	66.42
ProGen	65.94	57.28	59.49	38.70	33.10	36.72	35.97	69.30	57.90	63.57
S3	66.02	58.30	59.75	42.40	34.90	37.94	37.97	70.20	58.60	68.57
CLINGEN w/ KG	**74.85**	58.03	61.80	44.60	36.80	43.05	42.06	72.20	**65.80**	77.14
CLINGEN w/ LLM	72.40	**64.44**	**64.89**	**48.50**	**40.60**	**44.50**	**42.32**	**73.30**	61.30	**77.85**
CLINGEN w/ KG+LLM	75.10	64.12	65.81	50.57	40.65	40.60	39.59	68.30	66.70	77.85
PubMedBERT$_{\texttt{Large}}$										
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	31.72	30.50	58.10	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	77.36	**64.69**	**69.46**	**52.96**	**43.31**	46.05	**44.12**	**76.20**	66.80	**80.00**
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 PubMedBERT$_{\texttt{Base}}$ and PubMedBERT$_{\texttt{Large}}$.

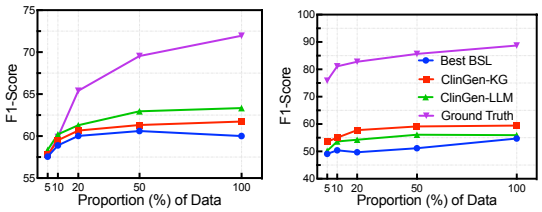
	BC5CDR-Disease			BC5CDR-Chemical			NCBI-Disease			CHEMDNER			CASI		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
PubMedBERT$_{\texttt{Base}}$															
Supervised-Full (SOTA)	—	—	86.10	—	—	93.33	—	—	88.76	—	—	92.35	—	—	—
Supervised-Full	83.84	87.92	85.83	92.22	91.74	91.98	87.54	89.92	88.71	91.84	92.45	92.14	—	—	—
Supervised-Few	24.86	39.47	30.51	63.73	46.07	53.48	36.16	39.47	37.74	48.00	28.70	35.92	38.11	43.82	40.77
DA-Word Sub	35.34	39.54	37.32	63.13	52.52	57.34	53.40	36.70	43.50	47.45	33.15	39.03	40.25	47.65	43.64
DA-Mixup	36.13	42.90	39.23	66.43	50.54	57.41	56.57	26.48	36.07	52.40	27.53	36.10	42.37	48.96	45.43
LightNER	39.80	33.20	36.20	—	—	—	43.70	41.90	42.78	—	—	—	—	—	—
DA-MELM	34.20	41.30	37.42	47.23	72.81	57.29	36.90	48.50	41.91	39.33	45.95	42.38	37.82	44.28	40.80
KGPC	50.80	51.30	51.05	—	—	—	52.20	52.10	52.15	—	—	—	—	—	—
ZeroGen	55.60	39.10	45.91	73.20	82.85	77.73	56.25	45.98	50.60	**54.34**	52.93	53.63	52.80	49.53	51.11
DemoGen	63.10	48.44	54.81	76.40	81.65	78.94	57.65	49.08	53.02	54.00	53.77	53.88	58.15	56.84	57.49
ProGen	61.60	50.50	55.50	77.10	82.02	79.48	56.01	53.50	54.73	51.55	53.00	52.26	57.76	58.57	58.16
S3	58.26	55.96	57.08	77.28	80.80	79.00	56.39	49.34	52.62	48.53	57.79	52.75	56.21	63.60	59.68
CLINGEN w/ KG	58.64	**63.02**	60.75	74.96	**85.45**	79.86	**62.62**	**56.62**	**59.47**	48.33	**69.28**	**56.94**	**71.75**	65.20	**68.32**
CLINGEN w/ LLM	**63.41**	58.83	**61.03**	**77.68**	84.33	**80.87**	62.58	50.59	55.95	51.40	58.77	54.84	68.19	**66.79**	67.48
CLINGEN w/ KG+LLM	60.57	66.21	63.26	73.66	87.30	79.90	58.01	65.37	59.17	52.07	63.62	57.27	72.57	70.48	71.51
PubMedBERT$_{\texttt{Large}}$															
Supervised-Full (SOTA)	—	—	86.39	—	—	94.04	—	—	89.18	—	—	92.72	—	—	—
Supervised-Full	86.77	85.92	86.34	92.80	92.94	92.87	87.97	90.09	89.02	92.23	92.48	92.35	—	—	—
Supervised-Few	25.52	45.85	32.79	61.40	54.41	57.69	44.86	40.12	42.35	43.40	34.60	38.50	41.30	45.02	43.08
DA-Word Sub	38.54	38.85	38.69	64.85	53.96	58.91	52.59	45.35	48.70	44.85	36.69	40.36	46.77	43.52	45.09
DA-Mixup	36.27	46.67	40.82	67.63	54.15	60.14	55.64	38.06	45.20	45.51	36.66	40.61	41.25	52.09	46.04
LightNER	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
DA-MELM	33.40	41.61	37.06	53.80	66.71	59.56	44.20	57.40	49.94	36.40	47.41	41.18	43.36	45.78	44.54
KGPC	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ZeroGen	57.40	39.21	46.59	78.08	80.97	79.49	54.52	49.00	51.61	48.56	59.44	53.45	54.04	51.40	52.69
DemoGen	57.34	49.48	53.12	78.27	83.90	80.99	59.43	56.83	58.10	48.03	60.39	53.51	62.67	61.02	61.83
ProGen	60.34	54.13	57.07	**78.42**	82.94	80.62	60.02	55.28	57.55	50.40	59.64	54.63	57.21	63.70	60.28
S3	65.46	51.86	57.87	77.89	84.31	80.97	56.00	53.49	54.72	54.80	53.88	54.33	63.07	62.72	62.89
CLINGEN w/ KG	54.28	**70.14**	61.21	77.88	86.32	81.88	**62.46**	**64.08**	**63.26**	47.03	**67.86**	**55.56**	70.96	**69.66**	**70.30**
CLINGEN w/ LLM	**61.05**	65.40	**63.15**	78.08	**86.98**	**82.29**	61.12	60.16	60.64	**50.92**	60.67	55.37	**71.61**	66.86	69.15
CLINGEN w/ KG+LLM	65.67	66.22	65.94	75.89	87.61	81.33	65.70	59.22	62.31	52.49	65.07	58.11	73.21	69.30	71.20

Table 9: Performance on token-classification tasks evaluated by PubMedBERT$_{\texttt{Base}}$ and PubMedBERT$_{\texttt{Large}}$.



(a) CDR (b) NCBI-Disease

Figure 10: Different generators at Base.

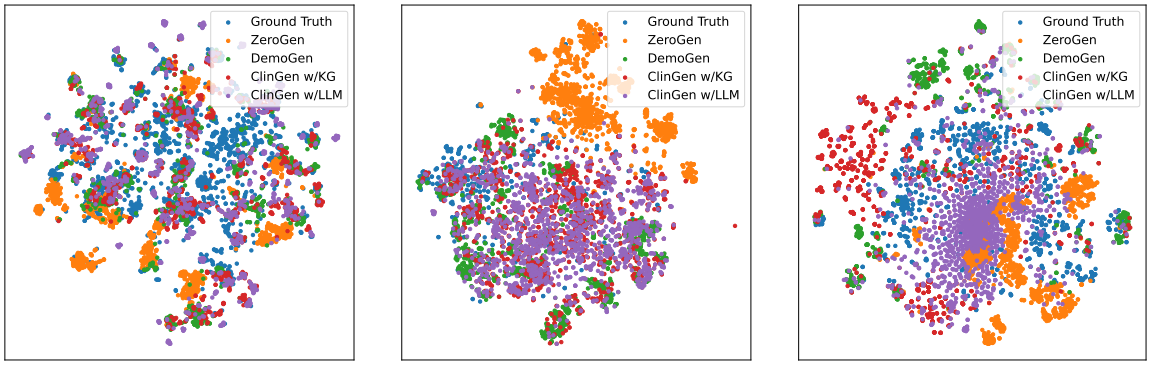


(a) CDR (b) NCBI-Disease

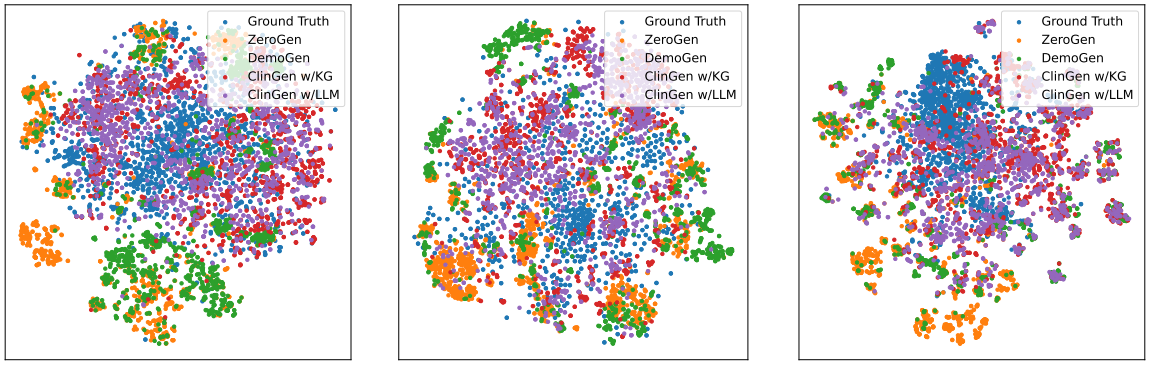
Figure 11: Different proportion of data at Base.

	HOC			CDR			MEDIQA-RQE			NCBI-Disease		
	Best Baseline	CLINGEN-KG	CLINGEN-LLM	Best Baseline	CLINGEN-KG	CLINGEN-LLM	Best Baseline	CLINGEN-KG	CLINGEN-LLM	Best Baseline	CLINGEN-KG	CLINGEN-LLM
1	70.04	74.30	77.30	61.52	61.66	63.34	68.30	76.85	74.50	56.12	60.22	54.51
2	75.30	79.73	73.63	60.69	63.77	64.66	64.20	71.80	71.19	54.19	60.64	57.81
3	71.41	74.81	78.33	57.82	59.79	62.02	67.18	75.90	71.51	53.85	57.52	55.50

Table 10: Performance with Different Random Seeds using PubMedBERT$_{\texttt{Base}}$.

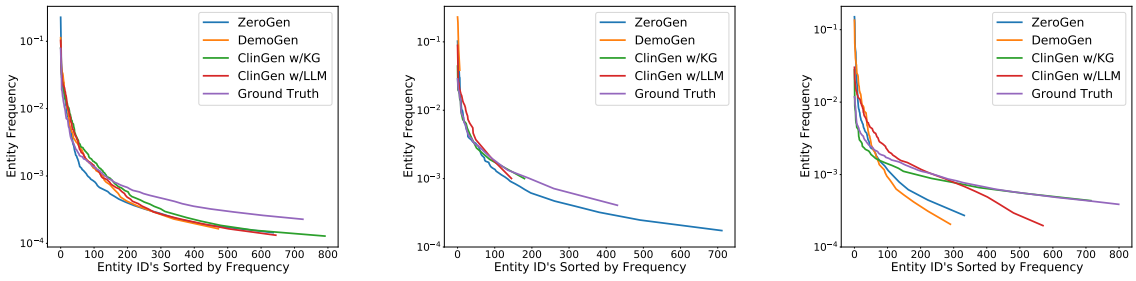


(a) LitCovid (b) GAD (c) CDR

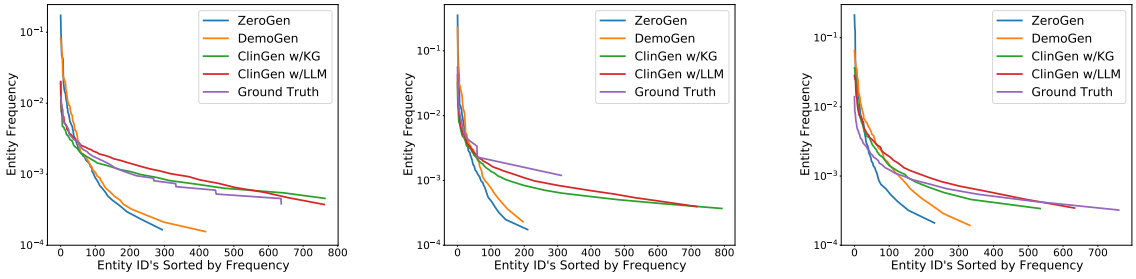


(d) MEDIQA-RQE (e) MQP (f) CHEMDNER

Figure 12: The t-SNE plots of datasets generated by CLINGEN, ZeroGen and DemoGen compared with the ground truth.



(a) LitCovid (b) GAD (c) CDR



(d) MEDIQA-RQE (e) MQP (f) CHEMDNER

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 outperform 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.
```

	LitCovid	CDR	MEDIQA-RQE	MQP	CHEMDNER	BC5CDR-Disease	Average
	F1	F1	ACC	ACC	F1	F1	–
PubMedBERT$_{\text{Base}}$							
Reframe (Mishra et al., 2022)	56.74	57.27	61.92	67.60	54.61	59.17	59.55
APE (Zhou et al., 2023)	56.24	61.12	66.55	68.00	52.10	58.79	60.47
PromptAgent (Wang et al., 2024)	56.62	48.44	63.64	61.00	54.47	59.98	57.36
CLINGEN w/ KG	58.01	61.75	**74.85**	72.20	**56.94**	60.75	**64.08**
CLINGEN w/ LLM	**59.22**	**63.34**	72.40	**73.30**	54.84	**61.03**	64.02
PubMedBERT$_{\text{Large}}$							
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.
Your task is to give a pair of
sentences with the following
instructions:
(1) Generate two sentences that
exhibit a clear similarity or
dissimilarity in meaning without
using complex or specialized
terms.
(2) express attributes
affirmatively.
(3) Ensure that both sentences
have a common attribute for
comparison.
(4) The length should be in 50
words.
```

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.
```

Listing 22: Prompt Format for generating sentences in text classification tasks with PromptAgent.

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

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:
(1) Generate a sentence that
contains a named entity. The
named entity should be a
```

```
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[8], 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.

	LitCovid	CHEMDNER
PubMedBERT$_{\text{Base}}$		
CLINGEN w/ KG	58.01	**56.94**
CLINGEN w/ LLM (ChatGPT)	**59.22**	54.84
CLINGEN w/ LLM (MedAlpaca)	55.45	52.15
PubMedBERT$_{\text{Large}}$		
CLINGEN w/ KG	55.81	**55.56**
CLINGEN w/ LLM (ChatGPT)	**57.07**	55.37
CLINGEN w/ LLM (MedAlpaca)	53.90	52.67

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.

KG : LLM	LitCovid	CDR	MEDIQA-RQE	BC5CDR-Disease	Average
	F1	F1	ACC	F1	–
PubMedBERT$_{\text{Base}}$					
1:0	58.01	61.75	74.85	60.75	63.84
2:1	56.18	62.89	73.50	60.53	63.28
1:1	56.76	63.86	74.01	63.26	64.47
1:2	55.49	64.33	75.10	61.62	64.14
0:1	59.22	63.34	72.40	61.03	64.00
PubMedBERT$_{\text{Large}}$					
1:0	55.81	62.66	79.92	61.21	64.90
2:1	54.21	64.22	76.15	62.40	64.25
1:1	56.80	65.90	79.12	65.94	66.94
1:2	54.41	64.68	80.77	64.55	66.10
0:1	57.07	64.99	77.36	63.15	65.64

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

[8] https://sites.research.google/med-palm/