Enrich, Aggregate, and Generate: Three-stage Biomedical Data-to-Text Generation Using Large Language Models in Low-resource Scenarios

Yupian Lin[1], Guangya Yu[1], Cheng Yuan[1], Hui Luo[1], Yuang Bian[2], Tong Ruan[1,*]
[1]East China University of Science and Technology, Shanghai 200237, China
[2]Zhongshan Hospital, Fudan University, Shanghai 200032, China
yupianlin@aliyun.com, ruantong@ecust.edu.cn

Abstract

Biomedical data-to-text generation aims at generating textual natural language descriptions that can fluently and precisely describe the biomedical structured data. However, biomedical data-to-text generation faces the dilemma of a lack of labeled data due to the privacy and scarcity of medical data. Large language models (LLMs) have demonstrated the ability to solve few-shot tasks through in-context learning (ICL). In this paper, we are the first to explore the performance of different LLMs in the biomedical data-to-text generation task. To address the issues of semantic sparsity and misinterpretation of numerical values in biomedical structured data, we propose an EAG (Enrich, Aggregate, and Generate) framework, a simple but efficient LLM-based three-stage biomedical D2T approach in low-resource scenarios. We conduct extensive evaluations of closed-source general LLMs, open-source general LLMs, and open-source medical LLMs. The results show that EAG framework provides good interpretability and superior performance, achieving state-of-the-art performance on the BioLeaflets dataset. The code and data will be released at https://github.com/FXLP/EAG.

1 Introduction

Data-to-text generation (D2T) (Lin et al., 2024) is an essential branch of Natural Language Generation (NLG), aiming at generating textual natural language descriptions that can fluently and precisely describe the structured data. Data-to-text generation has a wide variety of application scenarios in biomedicine, such as producing accurate and reliable assay validation reports (AVR), toxicology reports (Wu et al., 2022), clinical cases (Boulanger et al., 2024), and package leaflets of medicinal products (Yermakov et al., 2021).

In recent years, supervised natural language generation models have shown the ability to generate natural language text at an astounding degree of fluency and coherence, due to the advent of pre-trained language models (PLMs) such as BART (Lewis et al., 2020) and T5 (Raffel et al., 2020). However, biomedical data-to-text generation (especially in high-risk scenarios such as drug safety) faces the dilemma of a lack of labeled data due to the privacy and scarcity of medical data. To address this concern, researchers are exploring alternative methods in low-resource scenarios. Fortunately, large language models (LLMs) that contain hundreds of billions (or more) of parameters can solve few-shot tasks through in-context learning (ICL), which incorporates input-output demonstrations into the prompt.

However, to the best of our knowledge, existing research has not explored the performance of LLMs in biomedical data-to-text generation tasks. In this paper, we unprecedentedly use LLMs to generate medicinal product descriptions with only a few-shot examples. Unfortunately, the data construction method of the medicinal product information suffers from severe semantic sparsity, where the extracted entity list omits meaningful information from the original text. Therefore, we found that using LLMs directly for end-to-end biomedical data-to-text generation has many drawbacks, such as the inability to accurately and completely describe the content and the tendency to misinterpret numerical values, often mistakenly assigning them to unexpected sentences.

To address these high-risk challenges and data scarcity, we proposed the EAG (Enrich, Aggregate, and Generate) framework, a simple but efficient LLM-based three-stage biomedical D2T approach in low-resource scenarios. Specifically, the first stage of our proposed EAG framework is to enrich the semantics of structured inputs. Then this framework explicitly performs micro-planning in stage 2 before the text generation process, improving the interpretability of LLMs while taking advantage of

*Corresponding authors

Findings of the Association for Computational Linguistics: ACL 2026, pages 37611–37622

LLMs' powerful in-context learning capability.

The contributions of this paper are summarized as follows:

- We are the first to explore the effectiveness of different LLMs in the biomedical data-to-text generation task.
- We propose an EAG (Enrich, Aggregate, and Generate) framework, a simple but efficient LLM-based three-stage biomedical D2T approach to address the issues of semantic sparsity and misinterpretation of numerical values in biomedical structured data in low-resource scenarios.
- We conduct extensive evaluations of closed-source general LLMs, open-source general LLMs, and open-source medical LLMs. The results show that our proposed EAG framework provides good interpretability and superior performance, outperforming various baseline methods.
- We have constructed and publicly released a multi-version semantic-enhanced dataset (Enriched BioLeaflets) to facilitate research in this field.

The rest of the paper is organized as follows: Section 2 reviews the related work. Section 3 presents the details of our EAG framework. Section 4 shows experimental results and the case study, followed by a conclusion in Section 5.

2 Related Work

2.1 Biomedical Data-to-Text Generation

Data-to-text generation (D2T) (Lin et al., 2024, 2026, 2025, 2022) aims to generate textual natural language statements that fluently and precisely describe the structured data, such as graphs, tables, and meaning representations (MRs) in key-value pairs. In natural language processing, the data-to-text generation task has been a hotbed of research for many years. Significantly, structured data in the biomedical domain contains valuable information about individuals, treatments, and medications. Therefore, the research significance of biomedical data-to-text generation tasks is extraordinary. To produce accurate and reliable assay validation reports (AVR) and toxicology reports in medical table-to-text generation, Wu et al. (2022) proposed a novel two-step architecture containing the table extractor and the table generator, which is enhanced by auto-correction, copy mechanism, and synthetic data augmentation. TS-MRGen (Nishino et al., 2020) introduced a data-to-text generation module to the image diagnosis module for controllable medical report generation, improving the correctness and conciseness of generated reports. Boulanger et al. (2024) fine-tuned several pre-trained language models (PLMs) with different architectures (encoder-decoder and decoder-only) on French clinical cases, then generated clinical cases conditioned by patient demographic information (gender and age) and clinical features. Due to the privacy and scarcity of medical data (Amin-Nejad et al., 2020), there are few publicly available biomedical D2T datasets. Therefore, Yermakov et al. (2021) released a new real-world dataset (BioLeaflets) for benchmarking data-to-text generation models in the biomedical domain. They also presented baseline results of fine-tuning the PLMs such as BART (Lewis et al., 2020) and T5 (Raffel et al., 2020). In this paper, we are specifically interested in understanding various LLMs' capabilities on the biomedical D2T task.

2.2 Large Language Models in Medicine

Large language models (LLMs) have received a great deal of attention due to their exceptional human language understanding and generation abilities (Bian et al., 2025; Xue et al., 2025). Thus, applying LLMs to medicine to assist doctors and patients appears to be a promising area of research in both artificial intelligence and medicine (Thirunavukarasu et al., 2023; Yu et al., 2025; Hou et al., 2025). Existing medical LLMs (Zhou et al., 2023) are primarily constructed through the following four methods: (1) pre-training from scratch with a large amount of unsupervised medical corpus, such as GatorTronGPT (Peng et al., 2023); (2) continual pre-training based on general LLMs, such as MEDITRON (Chen et al., 2023b), Zhongjing (Yang et al., 2024b), PMC-LLaMA (Wu et al., 2023), ChiMed-GPT (Tian et al., 2023) and Qilin-Med (Ye et al., 2023); (3) fine-tuning from existing general LLMs, such as PULSE (Xiaofan Zhang, 2023), Baize (Xu et al., 2023), BenTsao (Wang et al., 2023b), BianQue (Chen et al., 2023a), ClinicalGPT (Wang et al., 2023a), CPLLM (Shoham and Rappoport, 2023) and Med-Gemini (Saab et al., 2024); (4) obtaining directly by prompting to align general LLMs to the medical domain, such as MedPrompt (Nori et al.,

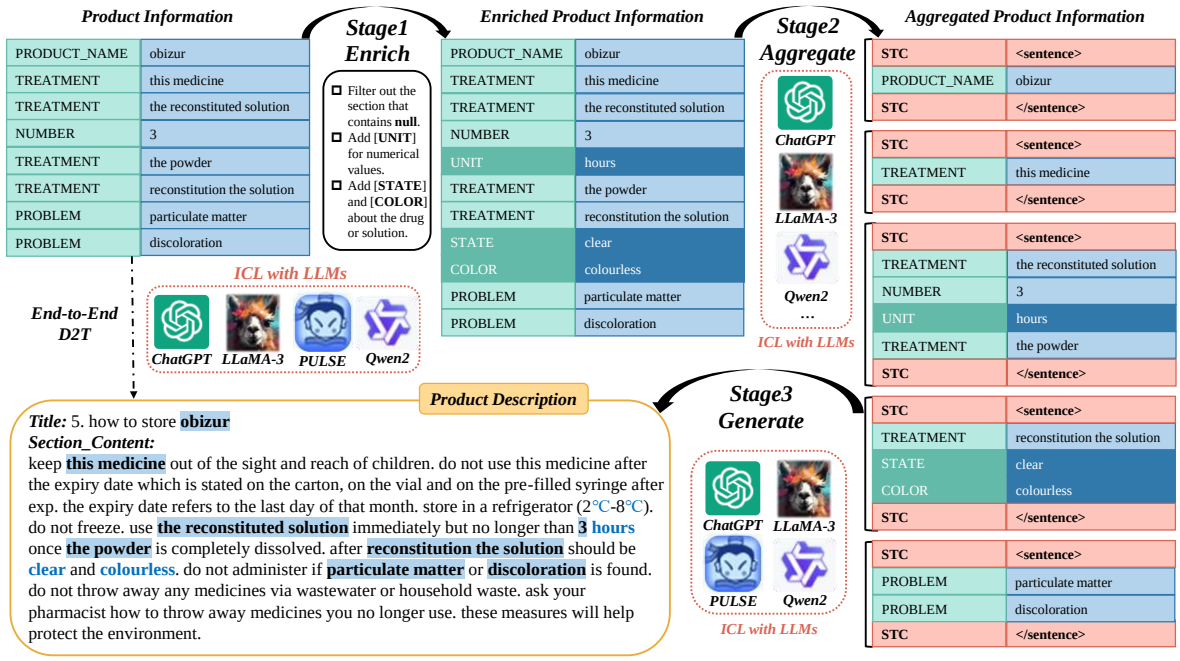

Figure 1: Overview of our EAG framework.

2023) and DeID-GPT (Liu et al., 2023). Continuing pre-training and fine-tuning based on general LLMs has become the most efficient and low-cost training paradigm for domain-specific LLMs. The commonly used medical data for training medical LLMs include multi-turn doctor-patient dialogues, medical exam Q&A, medical knowledge graphs, medical literature, medical books, Wikipedia, and electronic health records (EHRs). Furthermore, medical LLMs are typically evaluated on two popular types of downstream tasks: (1) discriminative tasks, which include question answering, entity extraction, relation extraction, text classification, natural language inference, semantic textual similarity, and information retrieval; (2) generative tasks, which include medical text summarization (Mathur et al., 2023; Nair et al., 2023), medical text generation (Milintsevich and Agarwal, 2023) and biomedical text simplification (Ondov et al., 2022). However, to our knowledge, existing research has not explored the performance of these LLMs in biomedical data-to-text generation tasks.

3 Methodology

In this section, we present our proposed **EAG** framework (as shown in Figure 1), an effective LLMs-based three-stage (**E**nrich, **A**ggregate, and **G**enerate) biomedical D2T approach. The prompt format used for EAG framework is shown in Figure 4 in Appendix B.

3.1 Stage 1: Enrich

As shown in Figure 1, the product information for medicinal products is structured data in the form of key-value pairs. The keys in key-value pairs are actually entity types in the NER process of constructing the BioLeaflets dataset. However, this data construction method suffers from severe semantic sparsity, where the extracted entity list omits meaningful information from the original text. To alleviate this issue, the first stage of our proposed EAG method is to enrich the semantics of structured inputs. Specifically, we constructed a multi-version semantic-enhanced BioLeaflets dataset through the following steps:

(1) Clean the original BioLeaflets dataset. In the first step, we filter out the section that contains ***null***. Table 1 shows the number of samples filtered out for each section type. We found that the majority of samples with null values appeared in Section 5 (How to store the product), totaling 311.

(2) Enrich the semantics of the cleaned BioLeaflets. In the second step, we mainly enrich the overall semantics of key-value pairs by adding two types of important information from the original product description: (a) *UNIT* for numerical values; (b) *STATE* and *COLOR* about the drug or solution.

(3) Divide entity sets for different sentences.

Section type	Train	Valid	Test	Total	Original total	Null
1. What the product is and what it is used for	1049	129	133	1311	1314	3
2. What you need to know before you take the product	1046	130	133	1309	1309	0
3. How to take the product	1044	133	133	1310	1313	3
4. Possible side effects	1028	130	129	1287	1295	8
5. How to store the product	682	98	81	861	1172	**311**
6. Content of the pack and other information	1042	129	133	1304	1311	7

Table 1: Enriched BioLeaflets dataset statistics grouped by section type.

In the traditional approach of D2T, micro-planning aims to convert a content plan into a sequence of sentences or phrases. It usually consists of three modules: aggregation, lexicalization, and referring expression generation. Aggregation is the process of grouping selected structured data together into sentences. In the third step, to construct demonstration examples for the second stage, we divide the ordered key-value pairs into sentences and insert special tokens (<sentence> and </sentence>) at the beginning and end of each sentence.

(4) Select demonstration examples for ICL. LLMs were shown to be able to adapt to new tasks by only seeing a few examples of the new task in their input, as opposed to needing additional training data or fine-tuning. This is typically referred to as In-Context Learning (ICL). LLMs generally benefit from more demonstrations, but as the number of demonstrations increases, the rate of improvement typically decreases (Xu et al., 2024). One barrier to increasing the number of demonstrations is the maximum context size of the LLM. As shown in the implementation details of the fourth section, the minimum context window for various LLMs in this paper is 8k. For a model with an 8k context window, the input can only contain up to 3 demonstration examples in this biomedical D2T task. As shown in Figure 2, by applying the k-means clustering technique, we categorize all demonstrations into k sub-groups (k = 3), aiming to group similar demonstrations. We pick the most representative demonstration from each sub-group, resulting in a final set of k demonstrations.

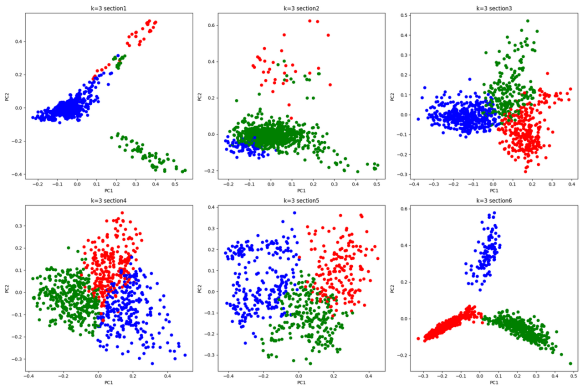

Figure 2: Demonstration Examples Selection by clustering (k=3) for 6 sections.

3.2 Stage 2: Aggregate

Neural modular approaches to D2T have proven that introducing a content planning stage before text generation can effectively enhance controllability and faithfulness (Lin et al., 2024). Therefore, our EAG framework added an Aggregate stage before the third stage (Generate) to explicitly teach the model micro-planning. As shown in Figure 1, the input structured data in this stage is enriched product information, and the output is aggregated product information, which is composed of multiple sets of key-value pairs belonging to different sentences. We will add k demonstration examples in the input prompts of the LLMs for In-Context Learning. Through this stage, the model can better assign each numerical value to the appropriate sentence and determine the actual meaning that each numerical value represents. It has partially reduced the ambiguity brought forth by the same number.

3.3 Stage 3: Generate

As shown in Figure 1, the Generate stage aims to convert aggregated product information into natural language text that describes input accurately and fluently. In this stage, we will also add k demonstration examples in the input prompts of the LLMs for In-Context Learning. The input of the demonstration example is aggregated product information, and the output is the product description. The prompt format used in this stage is also shown in Figure 4. Essentially, it is equivalent to executing the three modules in the traditional pipeline architecture, including lexicalization, referring expression generation, and linguistic realization.

4 Experiments and Analysis

In this section, we describe our experimental setup and report the evaluation results. Through comparative analysis with other baselines, we demonstrate the efficacy of our framework in the field of biomedical data-to-text generation.

4.1 Datasets

BioLeaflets (Yermakov et al., 2021) is a publicly available biomedical dataset for D2T generation, a corpus of 1336 package leaflets of medicines authorized in Europe. Package leaflets, which come in the packaging of medicinal products, provide patients with instructions on how to use the products safely and appropriately under the guidance of their healthcare professional. Package leaflets must be written clearly and straightforwardly. There are six sections in every document (see Table 1). This dataset is randomly split into training (80%), development (10%), and test (10%) sets. BioLeaflets proposes a conditional generation task: given an ordered set of entities as the source, the goal is to produce a multi-sentence section. It presents several challenges for the D2T task, including syntax, specialist medical vocabulary, a small sample size, and target text that is multi-sentence and multi-sectional.

4.2 Evaluation Metrics

We evaluate the generated description text from the following two aspects:

(1) We first assessed the informativeness of the generated texts using SacreBLEU (Post, 2018), ROUGE-L (Lin, 2004), and METEOR (Lavie and Agarwal, 2007), which measure lexical similarity by calculating the overlap of n-gram at the word level between the generated texts and the ground-truth descriptions.

(2) We second computed the contextual embedding-based metrics BERTScore (Zhang et al., 2020), MoverScore (Zhao et al., 2019), and BLEURT (Sellam et al., 2020) to evaluate the semantic similarity between the generated texts and the ground-truth descriptions.

4.3 Baselines

In these experiments, we mainly take into account the following baseline models[1].

(1) Non-pre-trained Models

- **Content Planner:** an LSTM-based neural D2T network architecture (Puduppully et al., 2019) that incorporates content selection and planning without sacrificing end-to-end training. Following Yermakov et al. (2021), we only use the content planning stage (encoder-decoder architecture with an attention mechanism) since only relevant entities are input to the model.

(2) Pre-trained Language Models (PLMs)

- **T5**: a text-to-text transfer transformer model (Raffel et al., 2020). We use the same hyperparameters reported by Yermakov et al. (2021): constant learning rate of 0.001, batch size of 32, 20 epochs, and greedy search as a decoding method.

- **BART**: a denoising autoencoder (Lewis et al., 2020) for pre-training sequence-to-sequence models with transformers. Following Yermakov et al. (2021), we use the same hyperparameters as per T5 fine-tuning.

- **BART and T5 with conditioning:** following Yermakov et al. (2021), we also add the prefix "section n" (n = 1, 2, . . . , 6) to the linearized input data. By doing this, the section type precondition for text generation is enforced, and the model is explicitly provided with information about the section number.

(3) Large Language Models (LLMs)

This family of LLMs contains tens or hundreds of billions of parameters. In this paper, we also add a baseline method that directly uses the following LLMs to accomplish the biomedical data-to-text generation task in a zero-shot manner.

- **Closed-source general LLMs:** we choose two models in ChatGPT: *gpt-4o-2024-05-13* and *gpt-4o-mini-2024-07-18*.

- **Open-source general LLMs:** we choose three powerful open-source general LLMs, such as LLaMA-3-8B (Meta, 2024), Qwen2-7B-Chat (Yang et al., 2024a), and InternLM-20B-Chat (Team, 2023).

- **Open-source medical LLMs:** we specifically selected two models, Llama-3-Physican-8B-Instruct (Guo et al., 2024) and PULSE-20B (Xiaofan Zhang, 2023), which were further

[1]To ensure a fair comparison, all models were evaluated under identical hyperparameters unless otherwise stated.

Method	Word-overlap metrics			Semantic equivalence metrics		
	SacreBLEU	ROUGE-L	METEOR	BERTScore	MoveScore	BLEURT
Non-LLM Methods						
Content Planner	*27.78	39.32	-	0.214	0.591	-0.072
BART-base	08.76	42.73	-	*0.370	0.609	*0.268
BART-base+cond	08.73	42.60	-	0.369	0.608	*0.268
T5-base	18.68	47.22	-	0.363	0.620	0.255
T5-base+cond	18.63	*47.31	-	0.364	*0.621	0.256
Closed-source general LLMs						
GPT-4o zero-shot	21.45	47.14	38.19	0.746	0.619	-0.374
GPT-4o 3-shot ICL	30.52	55.05	54.15	0.824	0.659	-0.148
GPT-4o 3-shot EAG	***32.71**	***58.96**	***54.95**	***0.832**	***0.667**	***-0.115**
GPT-4o mini zero-shot	19.77	36.92	38.53	0.789	0.623	-0.089
GPT-4o mini 3-shot ICL	27.38	46.88	42.21	0.810	0.641	-0.070
GPT-4o mini 3-shot EAG	**31.30**	**53.47**	**44.72**	**0.831**	**0.654**	***-0.012**
Open-source general LLMs						
LLaMA-3-8B zero-shot	10.73	31.48	29.92	0.730	0.593	-0.293
LLaMA-3-8B 3-shot ICL	17.76	39.52	36.32	0.756	0.608	**-0.283**
LLaMA-3-8B 3-shot EAG	**21.53**	**40.71**	***46.54**	**0.774**	**0.615**	-0.301
InternLM-20B-Chat zero-shot	07.51	24.81	22.56	0.724	0.578	**-0.339**
InternLM-20B-Chat 3-shot ICL	10.90	29.42	26.01	0.713	0.583	-0.415
InternLM-20B-Chat 3-shot EAG	**13.19**	**34.41**	**28.02**	**0.726**	**0.590**	-0.367
Qwen2-7B-Chat zero-shot	10.72	34.07	28.65	0.721	0.595	-0.308
Qwen2-7B-Chat 3-shot ICL	23.57	40.23	38.16	0.738	0.614	-0.229
Qwen2-7B-Chat 3-shot EAG	***26.60**	***46.38**	**40.90**	***0.806**	***0.636**	***-0.065**
Open-source medical LLMs						
Llama-3-Physician-8B-Instruct zero-shot	06.92	23.18	24.59	0.686	0.566	-0.506
Llama-3-Physician-8B-Instruct 3-shot ICL	20.51	38.11	38.93	0.770	0.614	-0.249
Llama-3-Physician-8B-Instruct 3-shot EAG	***22.13**	***42.47**	***41.63**	**0.771**	***0.620**	**-0.247**
PULSE-20B zero-shot	09.31	27.08	27.12	0.741	0.590	-0.241
PULSE-20B 3-shot ICL	16.15	33.49	31.09	0.739	0.600	-0.283
PULSE-20B 3-shot EAG	**19.33**	**40.27**	**35.70**	***0.777**	**0.616**	***-0.178**

Table 2: Performance comparisons of the automatic evaluation on the BioLeaflets dataset. * denotes the best performance within the same type (non-LLMs, closed-source general LLMs, open-source general LLMs, open-source medical LLMs).

pre-trained and fine-tuned using medical domain data based on the general LLMs mentioned above (LLaMA-3-8B and InternLM-20B-Chat).

4.4 Implementation Details

(1) Closed-source General LLMs: The context window size for both *gpt-4o-2024-05-13* and *gpt-4o-mini-2024-07-18* is 128k. To minimize the impact of randomness during the experiment, we use a temperature of 0.01 without any frequency penalty and top-k=1. The complete experiment of *gpt-4o-2024-05-13* and *gpt-4o-mini-2024-07-18* consumed a total of 191.81 US dollars.

(2) Open-source General LLMs: The context windows for LLaMA-3-8B, InternLM-20B-Chat, and Qwen2-7B-Chat are 8k, 16k, and 32k, respectively. We use the same hyperparameters for these open-source models: temperature=0.01, penalty=1.02, max_new_tokens=2500, and top-k=1. Top-p is set to 0.9, 0.8, and 0.8 on LLaMA-3-8B, InternLM-20B-Chat, and Qwen2-7B-Chat, respectively. The total inference time for LLaMA-3-8B, InternLM-20B-Chat, and Qwen2-7B-Chat on H800 is 8h, 40h, and 13h, respectively.

(3) Open-source Medical LLMs: The context window and hyperparameter settings of Llama-3-Physion-8B-Instruct are the same as its base model (LLaMA-3-8B). The context window and hyperparameter settings of PULSE-20B are the same as its base model (InternLM-20B-Chat). The total inference time for Llama-3-Physion-8B-Instruct and PULSE-20B on H800 is 9h and 43h, respectively.

4.5 Main Results and Analysis

Table 2 presents the comparison of automatic evaluation results between EAG and other baselines on the BioLeaflets dataset. From Table 2, it can be seen that our proposed EAG framework is generally superior to other methods regarding the three word-overlap metrics (SacreBLEU, ROUGE-L, and METEOR) and two semantic equivalence metrics (BERTScore and MoverScore). In low-resource scenarios, the 3-shot EAG method is always superior to the ordinary ICL method and has a significant improvement over the zero-shot setting.

Method	Word-overlap metrics			Semantic equivalence metrics		
	SacreBLEU	ROUGE-L	METEOR	BERTScore	MoverScore	BLEURT
Closed-source general LLMs						
GPT-4o mini 0-shot EAG	19.77	36.92	38.53	0.789	0.623	-0.089
GPT-4o mini 1-shot EAG	28.67	51.68	42.84	0.822	0.647	-0.049
GPT-4o mini 2-shot EAG	30.11	53.32	44.01	0.828	0.652	-0.012
GPT-4o mini 3-shot EAG w/o Enrich	27.38	46.88	42.21	0.810	0.641	-0.070
GPT-4o mini 3-shot EAG w/o Aggregate	27.26	47.11	42.24	0.810	0.641	-0.067
GPT-4o mini 3-shot EAG	31.30	53.47	44.72	0.831	0.654	-0.012
GPT-4o mini 4-shot EAG	31.77	54.33	45.26	0.831	0.655	**-0.006**
GPT-4o mini 5-shot EAG	**31.98**	**54.41**	**45.32**	**0.833**	**0.656**	-0.008
Open-source general LLMs						
Qwen2-7B-Chat 0-shot EAG	10.72	34.07	28.65	0.721	0.595	-0.308
Qwen2-7B-Chat 1-shot EAG	21.60	43.66	36.55	0.787	0.623	-0.125
Qwen2-7B-Chat 2-shot EAG	24.55	45.86	39.25	0.793	0.630	-0.100
Qwen2-7B-Chat 3-shot EAG w/o Enrich	23.57	40.23	38.16	0.738	0.614	-0.229
Qwen2-7B-Chat 3-shot EAG w/o Aggregate	23.82	40.33	38.05	0.738	0.614	-0.223
Qwen2-7B-Chat 3-shot EAG	**26.60**	**46.38**	**40.90**	**0.806**	**0.636**	**-0.065**
Qwen2-7B-Chat 4-shot EAG	23.36	44.47	38.23	0.765	0.620	-0.224
Qwen2-7B-Chat 5-shot EAG	24.95	44.65	39.09	0.776	0.626	-0.187
Open-source medical LLMs						
PULSE-20B 0-shot EAG	09.31	27.08	27.12	0.741	0.590	-0.241
PULSE-20B 1-shot EAG	13.22	33.09	29.67	0.757	0.601	-0.213
PULSE-20B 2-shot EAG	14.69	35.24	30.22	0.726	0.593	-0.342
PULSE-20B 3-shot EAG w/o Enrich	16.15	33.49	31.09	0.739	0.600	-0.283
PULSE-20B 3-shot EAG w/o Aggregate	15.77	33.84	31.07	0.742	0.600	-0.280
PULSE-20B 3-shot EAG	19.15	**40.27**	**35.70**	**0.777**	**0.616**	**-0.178**
PULSE-20B 4-shot EAG	**20.49**	40.25	34.76	0.751	0.613	-0.258
PULSE-20B 5-shot EAG	16.08	35.77	32.09	0.746	0.603	-0.262

Table 3: Ablation experiments on the BioLeaflets dataset. w/o: without.

The 3-shot EAG method achieves state-of-the-art performance on the BioLeaflets dataset using powerful LLMs. We further analyze the experimental results through the following three perspectives:

(1) Few-shot ICL vs Full Fine-tuning. As the parameter size of the pre-trained model continues to increase, so does the cost of fine-tuning the pre-trained model, due to this way modifies all parameters of the pre-trained model. However, few-shot ICL with LLMs (such as Qwen2) can achieve competitive results with the full-parameter fine-tuned BART and T5. Therefore, ICL with LLMs is more suitable for low-resource scenarios, achieving low-cost and high-quality text generation.

(2) Few-shot EAG vs Few-shot ICL. As shown in Table 2, the performance of 3-shot EAG is almost superior to that of 3-shot ICL in all metrics, even in different LLMs. All closed-source and open-source LLMs utilizing the EAG framework have shown significant improvements in all word-overlap metrics. GPT-4o performs the best in closed-source LLMs, while Qwen2-7B-Chat performs the best in open-source LLMs. It is worth noting that the Qwen2-7B-Chat 3-shot EAG achieved 0.806 on BERTScore, which is more than twice that of the BART and T5 models. Overall, the product description produced by 3-shot EAG is more fluent, factual, and grammatically correct than those by 3-shot ordinary ICL.

(3) General LLMs vs Medical LLMs. To explore whether the use of medical domain data for continual pre-training (CPT) of general LLMs can improve the effectiveness of this D2T task, we chose two pairs of LLMs for comparison. Specifically, the first group compared LLaMA-3-8B and Llama-3 Physican-8B-Instruct, while the second group compared InternLM-20B-Chat and PULSE-20B. From Table 2, it can be seen that Llama-3 Physican-8B-Instruct outperforms LLaMA-3-8B in over half of the automatic evaluation metrics in the 3-shot EAG setting. Even more interesting is that PULSE-20B outperforms its base model, InternLM-20B-Chat, across all six automatic evaluation metrics. Therefore, we can conclude that continual pre-training (CPT) with medical domain data based on general LLMs can improve the model's performance on biomedical D2T tasks.

4.6 Ablation Study

Moreover, to verify the effectiveness of different modules, we compare EAG with its variants on three representative LLMs (GPT-4o mini, Qwen2-7B-Chat, and PULSE-20B) with at least a 16k context window, as an 8k context window can only

contain a maximum of 3 demonstration examples of BioLeaflets. Table 3 shows our ablation experimental results. We then further explore the factors that affect the performance of the LLMs based on the following four questions:

(1) Is few-shot better than zero-shot?

As shown in Table 3, the results of all few-shot EAG with demonstration examples are significantly better than those of zero-shot EAG. On three word-overlap metrics, few-shot EAG improved by over 6% in all LLMs. Under the few-shot setting, all LLMs improved by over 2% on all semantic equivalence metrics compared to zero-shot setting.

(2) Are more demonstration examples better?

To investigate the impact of the number of demonstration examples on models with different contextual window sizes, we tested the performance of three LLMs from 1-shot to 5-shot. As shown in Table 3, both Qwen2-7B-Chat (32k) and PULSE-20B (16k) achieved their best performance at 3-shot, while GPT-4o mini (128k) still showed a trend of better model performance with increasing examples in the context. Therefore, the model's ICL ability is limited by its contextual window. Providing more demonstration examples in the input can improve the model's performance for LLMs with a larger context window.

(3) Is the stage 1 (Enrich) effective?

To verify whether stage 1 (Enrich) is effective for the LLMs, we compared the results of 3-shot EAG w/o Enrich and 3-shot EAG. Table 3 shows that compared to 3-shot EAG, 3-shot EAG w/o Enrich exhibits significant performance degradation across all evaluation metrics. It indicates that stage 1 (Enrich) plays a crucial role in the EAG framework for different LLMs (GPT-4o mini, Qwen2-7B-Chat, and PULSE-20B).

(4) Is the stage 2 (Aggregate) effective?

To verify whether stage 2 (Aggregate) is effective for the different LLMs, we also compared the results of 3-shot EAG w/o Aggregate and 3-shot EAG. Interestingly, 3-shot EAG w/o Aggregate exhibits almost the same results as 3-shot EAG w/o Enrich. This indicates that removing either stage 1 or stage 2 from the EAG framework will result in a significant decline in the performance of the LLMs. Therefore, stage 2 (Aggregate) is also a decisive factor in EAG, as it explicitly prompts the model to perform correct content planning.

4.7 Case Study

To understand the effect of our method more intuitively, we select one representative example (*Afstyla*) and present its descriptions generated by different settings (3-shot ICL, 0-shot EAG, and 3-shot EAG) with the powerful GPT-4o mini model in Figure 3 in Appendix A. We can see that the GPT-4o mini model can learn the same schema from multiple demonstration examples under the 3-shot ICL setting, but its understanding of numerical values is still incorrect. In addition, although the GPT-4o mini model follows the aggregation instructions under the 0-shot EAG setting, it cannot accurately understand the task due to the lack of demonstration examples. It is satisfying that 3-shot EAG not only learns potential templates from the reference text through contextual examples (as shown in the first sentence of section 5, which is usually "Keep this medicine out of the sight and reach of children") but also correctly understands the meaning represented by the numbers ("3 months"). Overall, the case reveals that 3-shot EAG is the optimal solution.

5 Conclusion

In this paper, we are the first to explore the effectiveness of different large language models in the biomedical data-to-text generation task. To address the issues of semantic sparsity and misinterpretation of numerical values in structured data, we propose an EAG (Enrich, Aggregate, and Generate) framework, a simple but efficient LLM-based three-stage biomedical D2T approach. Experiments on different LLMs have shown that our EAG framework achieves SOTA performance on the BioLeaflets dataset with only three demonstration examples. In the future, we plan to continue pre-training more powerful open-source general LLMs with high-quality medical domain data to improve performance in the biomedical D2T task.

Limitations

Our approach has the following limitations: (1) The contextual examples chosen by k-means clustering are not necessarily the most appropriate, and there is still much room for improvement. (2) This method is still costly because it can only perform well based on large language models. Therefore, we need to think about how to give similar micro-planning and surface realization powers to smaller models in low-resource scenarios.

References

Ali Amin-Nejad, Julia Ive, and Sumithra Velupillai. 2020. Exploring transformer text generation for medical dataset augmentation. In *International Conference on Language Resources and Evaluation*.

Yuang Bian, Yupian Lin, Jingping Liu, and Tong Ruan. 2025. Ptoco: Prefix-based token-level collaboration enhances reasoning for multi-llms. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8326–8335.

Hugo Boulanger, Nicolas Hiebel, Olivier Ferret, Karën Fort, and Aurélie Névéol. 2024. Using structured health information for controlled generation of clinical cases in french. In *The 6th Clinical Natural Language Processing Workshop At NAACL 2024 (ClinicalNLP 2024)*.

Yirong Chen, Zhenyu Wang, Xiaofen Xing, Zhipei Xu, Kai Fang, Junhong Wang, Sihang Li, Jieling Wu, Qi Liu, Xiangmin Xu, et al. 2023a. Bianque: Balancing the questioning and suggestion ability of health llms with multi-turn health conversations polished by chatgpt. *arXiv preprint arXiv:2310.15896*.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, et al. 2023b. Meditron-70b: Scaling medical pretraining for large language models. *arXiv preprint arXiv:2311.16079*.

Yiduo Guo, Jie Fu, Huishuai Zhang, Dongyan Zhao, and Yikang Shen. 2024. Efficient continual pre-training by mitigating the stability gap. *arXiv preprint arXiv:2406.14833*.

Ruihui Hou, Shencheng Chen, Yongqi Fan, Guangya Yu, Lifeng Zhu, Jing Sun, Jingping Liu, and Tong Ruan. 2025. Msdiagnosis: A benchmark and framework for evaluating large language models in multi-step clinical diagnosis. *Knowledge-Based Systems*, 330:114524.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: an automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation, WMT@ACL 2007, Prague, Czech Republic, June 23, 2007*, pages 228–231. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, page 7871. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yupian Lin, Yuang Bian, Guangya Yu, Dongge Xue, Wanpeng Lu, Jingping Liu, and Tong Ruan. 2025. Cot-planner: Chain-of-thoughts as the content planner for few-shot table-to-text generation reduces the hallucinations from llms. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Yupian Lin, Tong Ruan, Ming Liang, Tingting Cai, Wen Du, and Yi Wang. 2022. Dotat: A domain-oriented text annotation tool. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–8.

Yupian Lin, Tong Ruan, Jingping Liu, and Haofen Wang. 2024. A survey on neural data-to-text generation. *IEEE Transactions on Knowledge and Data Engineering*, 36:1431–1449.

Yupian Lin, Guangya Yu, Cheng Yuan, Huan Du, Hui Luo, Yuang Bian, Jingping Liu, Zhidong He, Wen Du, and Tong Ruan. 2026. LogToP: Logic tree-of-program with table instruction-tuned LLMs for controlled logical table-to-text generation. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 5291–5303, Rabat, Morocco. Association for Computational Linguistics.

Zhengliang Liu, Yue Huang, Xiaowei Yu, Lu Zhang, Zihao Wu, Chao Cao, Haixing Dai, Lin Zhao, Yiwei Li, Peng Shu, et al. 2023. Deid-gpt: Zero-shot medical text de-identification by gpt-4. *arXiv preprint arXiv:2303.11032*.

Yash Mathur, Sanketh Rangreji, Raghav Kapoor, Medha Palavalli, Amanda Bertsch, and Matthew R. Gormley. 2023. Summqa at mediqa-chat 2023: In-context learning with gpt-4 for medical summarization. In *Clinical Natural Language Processing Workshop*.

AI Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI*.

Kirill Milintsevich and Navneet Agarwal. 2023. Calvados at mediqa-chat 2023: Improving clinical note generation with multi-task instruction finetuning. In *Clinical Natural Language Processing Workshop*.

Varun Nair, Elliot Schumacher, and Anitha Kannan. 2023. Generating medically-accurate summaries of patient-provider dialogue: A multi-stage approach using large language models. In *Clinical Natural Language Processing Workshop*.

Toru Nishino, Ryota Ozaki, Yohei Momoki, Tomoki Taniguchi, Ryuji Kano, Norihisa Nakano, Yuki Tagawa, Motoki Taniguchi, Tomoko Ohkuma, and Keigo Nakamura. 2020. Reinforcement learning with imbalanced dataset for data-to-text medical report generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2223–2236.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, et al.

2023. Can generalist foundation models outcompete special-purpose tuning? case study in medicine. *arXiv preprint arXiv:2311.16452*.

Brian Ondov, Kush Attal, and Dina Demner-Fushman. 2022. A survey of automated methods for biomedical text simplification. *Journal of the American Medical Informatics Association*, 29(11):1976–1988.

Cheng Peng, Xi Yang, Aokun Chen, Kaleb E Smith, Nima PourNejatian, Anthony B Costa, Cheryl Martin, Mona G Flores, Ying Zhang, Tanja Magoc, et al. 2023. A study of generative large language model for medical research and healthcare. *NPJ digital medicine*, 6(1):210.

Matt Post. 2018. A call for clarity in reporting bleu scores. In *Conference on Machine Translation*.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with content selection and planning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6908–6915.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Khaled Saab, Tao Tu, Wei-Hung Weng, Ryutaro Tanno, David Stutz, Ellery Wulczyn, Fan Zhang, Tim Strother, Chunjong Park, Elahe Vedadi, et al. 2024. Capabilities of gemini models in medicine. *arXiv preprint arXiv:2404.18416*.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Annual Meeting of the Association for Computational Linguistics*.

Ofir Ben Shoham and Nadav Rappoport. 2023. Cpllm: Clinical prediction with large language models. *arXiv preprint arXiv:2309.11295*.

InternLM Team. 2023. Internlm: A multilingual language model with progressively enhanced capabilities.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature Medicine*, 29:1930–1940.

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

Guangyu Wang, Guoxing Yang, Zongxin Du, Longjun Fan, and Xiaohu Li. 2023a. Clinicalgpt: large language models finetuned with diverse medical data and comprehensive evaluation. *arXiv preprint arXiv:2306.09968*.

Haochun Wang, Chi Liu, Nuwa Xi, Zewen Qiang, Sendong Zhao, Bing Qin, and Ting Liu. 2023b. Huatuo: Tuning llama model with chinese medical knowledge. *arXiv preprint arXiv:2304.06975*.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. Pmc-llama: Further finetuning llama on medical papers. *arXiv preprint arXiv:2304.14454*, 2(5):6.

Heng-Yi Wu, Jingqing Zhang, Julia Ive, Tong Li, Narges Tabari, Bingyuan Chen, Vibhor Gupta, and Yike Guo. 2022. Medical scientific table-to-text generation with human-in-the-loop under the data sparsity constraint. *CoRR*, abs/2205.12368.

Shaoting Zhang Xiaofan Zhang, Kui Xue. 2023. Pulse: Pretrained and unified language service engine.

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6268–6278.

Xin Xu, Yue Liu, Panupong Pasupat, Mehran Kazemi, et al. 2024. In-context learning with retrieved demonstrations for language models: A survey. *arXiv preprint arXiv:2401.11624*.

Dongge Xue, Zhili Pu, Zhentao Xia, Hongli Sun, Ruihui Hou, Guangya Yu, Yupian Lin, Yongqi Fan, Jingping Liu, and Tong Ruan. 2025. Text-to-es bench: A comprehensive benchmark for converting natural language to elasticsearch query. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19767–19790.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024a. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Songhua Yang, Hanjie Zhao, Senbin Zhu, Guangyu Zhou, Hongfei Xu, Yuxiang Jia, and Hongying Zan. 2024b. Zhongjing: Enhancing the chinese medical capabilities of large language model through expert feedback and real-world multi-turn dialogue. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19368–19376.

Qichen Ye, Junling Liu, Dading Chong, Peilin Zhou, Yining Hua, and Andrew Liu. 2023. Qilinmed: Multi-stage knowledge injection advanced medical large language model. *arXiv preprint arXiv:2310.09089*.

Ruslan Yermakov, Nicholas Drago, and Angelo Ziletti. 2021. Biomedical data-to-text generation via fine-tuning transformers. In *International Conference on Natural Language Generation*.

Guangya Yu, Yanhao Li, Zongying Jiang, Yuxiong Jin, Li Dai, Yupian Lin, Ruihui Hou, Weiyan Zhang, Yongqi Fan, Qi Ye, et al. 2025. Cmqcic-bench: A chinese benchmark for evaluating large language models in medical quality control indicator calculation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 609–626.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Conference on Empirical Methods in Natural Language Processing*.

Hongjian Zhou, Boyang Gu, Xinyu Zou, Yiru Li, Sam S. Chen, Peilin Zhou, Junling Liu, Yining Hua, Chengfeng Mao, Xian Wu, Zheng Li, and Fenglin Liu. 2023. A survey of large language models in medicine: Progress, application, and challenge. *ArXiv*, abs/2311.05112.

A The case of the BioLeaflets dataset

The representative example (*Afstyla*) and its descriptions generated by different methods (3-shot ICL, 0-shot EAG, and 3-shot EAG) with the GPT-4o mini model are shown in Figure 3.

B The prompt format of EAG

To provide readers with a more intuitive and clear understanding of how our proposed EAG framework is implemented, we offer the prompt format used for each setting in Figure 4.

C The Experimental Platform

Our experiments are conducted on a workstation running Ubuntu 20.04.6 LTS, with two Intel (R) Xeon (R) Platinum 8336C CPUs, four NVIDIA A800 GPUs, and 1.0TiB of memory.

D Human evaluation experiment

In order to better evaluate the quality of the generated descriptions, we recruited domain experts to conduct a human evaluation experiment on whether the numerical values in the generated descriptions are understood correctly. We randomly selected 100 samples from the test set as evaluation objects to assess the performance of the GPT-4o mini model under different settings (3-shot ICL, 0-shot EAG, and 3-shot EAG). The results of the human evaluation experiment are shown in Table 4.

Table 4 shows that the accuracy of 0-shot EAG in Sections 3, 4, and 6 has been significantly improved, while it has also achieved comparable results with 3-shot ICL in the other three Sections. The average accuracy (AVG) of 0-shot EAG is 18.6% higher than that of 3-shot ICL. The AVG of the 3-shot EAG further increased significantly, 44.3% higher than that of the 3-shot ICL, doubling the numerical understanding ability of the LLMs. This result further proves the effectiveness of our proposed EAG framework in the biomedical data-to-text generation task.

E Qualitative error analysis

To clarify the article, we have added a qualitative error analysis to illustrate where the large language model still fails under the EAG framework. Section 1 (*What the product is and what it is used for*) and Section 5 (*How to store the product*) of the medicinal products package leaflets generated by the 3-shot EAG method have been very accurate (reaching more than 90% in Table 4). The accuracy of Section 2 (*What you need to know before you take the product*) and Section 6 (*Content of the pack and other information*) of the package leaflets generated by 3-shot EAG is similar (both 82% in Table 4). Because the text length of Section 2 of the package leaflets exceeds 5000, the LLMs are prone to missing numerical information at the end of the structured input. In addition, Section 6 of the package leaflets contains a lot of numerical information about the specific composition and packaging details of medical products. Therefore, the LLMs may confuse different components. Sections 3 (*How to take the product*) and 4 (*Possible side effects*) of the package leaflets generated by the 3-shot EAG method still have some errors in numerical understanding (the accuracy rate is 73% in Table 4). After carefully reviewing the error examples, we found that the errors of the LLMs were concentrated in two places: on the one hand, the time and duration of medication (in Section 3); on the other hand, the prevalence of side effects, such as one tenth, one percent, and one thousandth (in Section 4). In the future, this will be an important direction to improve.

GPT-4o mini 3-shot ICL

Title: 5. how to store afstyla Keep this medicine out of the sight and reach of children. Do not use this medicine after the expiry date which is stated on the label and the carton. The expiry date refers to the last day of that month. Store in a refrigerator (2 - 8° C). Do not freeze. Keep the afstyla powder in the outer carton in order to protect from light. The reconstituted product can be stored at room temperature (not above 25° C) for a single period of up to 3 hours. Do not throw away any medicines via wastewater or household waste. Ask your pharmacist how to throw away medicines you no longer use. These measures will help protect the environment.

GPT-4o mini 0-shot EAG

5. How to Store Afstyla <sentence>Store Afstyla in the refrigerator at a temperature between 2° C and 8° C.</sentence><sentence>Do not freeze this medicine.</sentence><sentence>Keep the vial in the outer carton to protect it from light.</sentence><sentence>Once reconstituted, the product should be used within 3 hours.</sentence><sentence>Do not use the Afstyla powder after the expiry date which is stated on the label and carton. The expiry date refers to the last day of that month.</sentence><sentence>After reconstitution, the product can be stored at room temperature (up to 25° C) for up to 3 hours.</sentence><sentence>Do not store the reconstituted product in the refrigerator.</sentence><sentence>Keep this medicine out of the sight and reach of children.</sentence>

GPT-4o mini 3-shot EAG

keep this medicine out of the sight and reach of children. do not use this medicine after the expiry date which is stated on the label. the expiry date refers to the last day of that month. do not store above 8 ° C. the afstyla powder should be stored in the outer carton in order to protect from moisture. the reconstituted product can be stored at room temperature (not above 25° C) and should be used within 3 months. do not or household waste. ask your pharmacist how to throw away medicines you no longer use. these measures will help protect the environment. throw away any medicines via wastewater

Product Description

keep this medicine out of the sight and reach of children. do not use this medicine after the expiry date, which is stated on the label and carton. store in a refrigerator (2 ° C to 8 ° C). before the afstyla powder is reconstituted it may be kept at room temperature (below 25 ° C) for a single period not exceeding 3 months, within the expiration date printed on the cartons and the vials. please record the date from when you start to store afstyla at room temperature on the product carton. once the product has been taken out of the refrigerator, the product must not be returned to the refrigerator. do not freeze. keep the vial in the outer carton in order to protect from light. the reconstituted product should preferably be used immediately. if the reconstituted product is not administered immediately, storage times and conditions prior to use are in the responsibility of the user.

Figure 3: Descriptions generated by different settings (3-shot ICL, 0-shot EAG, and 3-shot EAG) with the powerful GPT-4o mini model and the section 5 (how to store afstyla) in product description of the *Afstyla*.

Method	Section 1	Section 2	Section 3	Section 4	Section 5	Section 6	AVG
3-shot ICL	100	27	18	36	45	9	39.2
0-shot EAG	100	27	46	55	46	73	57.8
3-shot EAG	100	82	73	73	91	82	83.5

Table 4: The results of the human evaluation experiment on the GPT-4o mini model. AVG represents the geometric mean of the numerical understanding accuracy of six sections.

Prompt Format of EAG

<<System Role Setting>>
Assume you are a doctor writing a drug information leaflet. Below, I will provide you with the title of the leaflet and some sorted key-value pairs describing the content of the leaflet.
<<Instruction for Aggregate>>
Please determine which key-value pair information originally belonged to the same sentence, and add <sentence> at the beginning and </sentence> at the end of the group of key-value pairs that form the same sentence.
<<Instruction for Generate>>
Please use these key-value pair information to restore the original drug information leaflet.
<<Instruction for In-Context Learning>>
Here are some examples: {ICL_Content}. Please refer to the samples to process the following data.
Title: {Title}. Sorted key-value pairs: {Key-Value Pairs}. Restored leaflet content:

Figure 4: The prompt format of EAG.