GOLD: Generalized Knowledge Distillation via Out-of-Distribution-Guided Language Data Generation

Mohsen Gholami[1,2], **Mohammad Akbari**[1], **Cindy Hu**[1], **Vaden Masrani**[1],
Z. Jane Wang[2], and **Yong Zhang**[1]
[1]Huawei Technologies Canada Co. Ltd.
[2]University of Biritish Columbia
{mohsen.gholami, mohammad.akbari, cindy.hu1, yong.zhang3}@huawei.com

Abstract

Knowledge distillation from LLMs is essential for the efficient deployment of language models. Prior works have proposed data generation using LLMs for preparing distilled models. We argue that generating data with LLMs is prone to sampling mainly from the center of original content distribution. This limitation hinders the distilled model from learning the true underlying data distribution and to forget the tails of the distributions (samples with lower probability). To this end, we propose *GOLD* , a task-agnostic data generation and knowledge distillation framework, which employs an iterative out-of-distribution-guided feedback mechanism for the LLM. As a result, the generated data improves the generalizability of distilled models. An energy-based OOD evaluation approach is also introduced to deal with noisy generated data. Our extensive experiments on 10 different classification and sequence-to-sequence tasks in NLP show that *GOLD* respectively outperforms prior arts and the LLM with an average improvement of 5% and 14%. We will also show that the proposed method is applicable to less explored and novel tasks. Code is available here[1].

1 Introduction

Large language models (LLMs) have shown outstanding few-shot performance in solving different complex natural language tasks (Brown et al., 2020). The term few-shot refers to the ability of the LLM to understand and perform tasks accurately given only a few examples. However, achieving such performance necessitates models with a large number of parameters to generalize and learn distinct tasks. The computational complexity of LLMs hinders their real-world applications for deployment. Moreover, most of the LLMs are not publicly available and users should share their confidential data with LLMs through prompting which is a privacy concern. Therefore, user-specific small models are critical to address efficiency and privacy concerns.

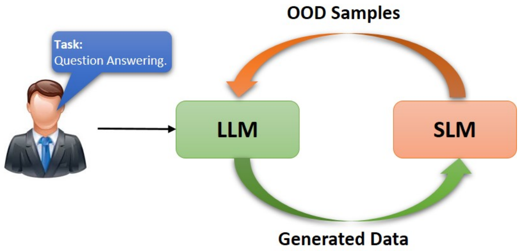

Figure 1: *GOLD* finds failure modes of SLM in the course of data generation and guides the LLM to generate OOD samples to improve SLM's generalizability.

To circumvent the above challenges, knowledge distillation (KD) from LLMs has been used to prepare small language models (SLMs). There are two main paradigms in KD: *data-informed* and *data-free* methods (Agarwal et al., 2023; Hsieh et al., 2023; Gu et al., 2023). Data-informed methods are the conventional KD techniques that use LLMs to label data and train a small language model with the labels (Agarwal et al., 2023; Hsieh et al., 2023). Data-free methods, on the other hand, study an extreme case when no human-generated dataset (unlabeled or labeled) is available during distillation. ZeroGen (Ye et al., 2022a) and ProGen (Ye et al., 2022b) are two prior works that proposed data-free knowledge distillation.

ZeroGen (Ye et al., 2022a) proposed generating data for classification tasks using LLMs and trained a small model on the generated data. ProGen (Ye et al., 2022b) improved the quality of the generated data by finding important samples via an influence function in the course of data generation. These works are not able to find failure modes of SLMs and steer data generation toward the samples that can improve the generalizability of the model.

(Shumailov et al., 2023) showed that distilling

[1] https://developer.huaweicloud.com/develop/aigallery/notebook/detail?id=9d770d1f-3758-4d0f-99d4-3346abbe1546

Findings of the Association for Computational Linguistics: NAACL 2024, pages 4365–4380
June 16-21, 2024 ©2024 Association for Computational Linguistics

knowledge from LLMs via data generation causes irreversible defects in the SLM, where the tails of the original content distribution disappear. LLMs usually tend to generate samples with higher likelihood (known as in-distribution samples) repeatedly (Shumailov et al., 2023) which results in poor generalizability of the distilled SLM. Preserving the LLM's ability to model low-probability events or out-of-distribution (OOD) samples is essential to the fairness of their predictions. Such events are often relevant to marginalized groups and are vital to understand complex systems (Taleb, 2007).

Usually, the performance of language models improves when more human-labeled data is available, since a larger dataset covers wider task domains. On the contrary, (Gudibande et al., 2023) shows that increasing the amount of generated data with LLMs lowers the performance of the distilled SLM. We argue that vanilla data generation with LLMs is prone to only generate high likelihood samples which negatively affects the SLM performance with increasing the number of samples.

To this end, we propose *GOLD*, a data-free KD framework that iteratively finds failure modes of the SLM (i.e., OOD samples) and provides feedback to the LLM (Figure 1) for the next iteration of data generation. Figure 2 shows the overall framework of *GOLD*. The user provides the task definition with a few samples of the data corresponding to that task. The LLM is then used to generate a training batch of data for the specified task, which is used to update the weights of the SLM. Given the generated train batch, we use prompting to ask LLM to generate a separate batch of OOD data that is significantly different from the train batch in terms of topic and style This OOD batch is then used as our validation set to evaluate the performance of the SLM and accordingly identify its failure modes as a feedback to the LLM. We use the output logits of the SLM to measure the energy values of the validation samples and select the top ones based on their corresponding free energy scores (Liu et al., 2020; Akbari et al., 2023). The energy score does not require the labels of the generated sample and therefore is not prone to pick the data with noisy labels as the OOD samples. The selected OOD samples are then used as the feedback to the LLM for the next iteration of data generation. The major contributions of this paper are:

- Proposing a task-agnostic framework for data generation and KD from LLMs to SLMs that is applicable to any NLP task, even novel tasks.
- Introducing an iterative OOD-empowered feedback mechanism to improve the generalizability of distilled SLMs.
- Proposing an energy-based OOD evaluation approach to handle noisy data generated by LLMs.
- Achieving state-of-the-art results on a variety of classification and sequence-to-sequence downstream tasks in NLP.

2 Related Works

Data-Informed KD. Distil-Step-by-Step (Hsieh et al., 2023) is a data-informed approach which proposes to perform KD from an LLM as an annotation function for real data, where the annotations take the form of "rationals" extracted via Chain-of-Thought prompting. Other data-informed methods such as MiniLLM (Gu et al., 2023) and GKD (Agarwal et al., 2023) also propose to change the training objective used for KD away from the commonly-used forward Kullback-Leibler divergence between the teacher and student distribution.

Data-Free KD. ZeroGen (Ye et al., 2022a) and ProGen (Ye et al., 2022b) are two data-free approaches to train task-specific student models with much fewer parameters than the base model. Using task-specific prompting in place of human annotations, and a sufficiently large generated train set, (Ye et al., 2022a) can *exceed* the performance of the base model on text classification, question answering, and natural language inference. ProGen (Ye et al., 2022b) then builds on ZeroGen by using an "influence function" to measure the quality of the generated samples on a synthetic validation set, and uses this to reduce the portion of synthetic data which is low-quality or redundant. ProGen achieves on-par or superior performance with only 1% synthetic dataset size compared to ZeroGen.

Limitations of Data Generation with LLMs. Two recent works (Gudibande et al., 2023; Shumailov et al., 2023) challenge the idea that training a generative model on the output of a different generative model will lead to performance gains in the case of a smaller "imitation models" being trained to match the performance of a "teacher model" as in (Gudibande et al., 2023). However, we note their negative findings only apply in the most general case, where the synthetic train data is not "corrected" after being generated, either algorithmically or manually by humans before being hosted

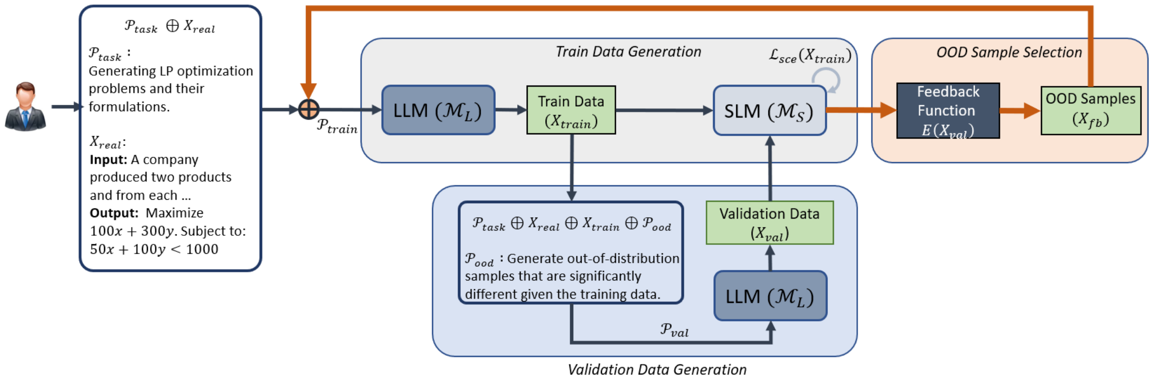
$\mathcal{P}_{task} \oplus X_{real}$
$\mathcal{P}_{task}$: Generating LP optimization problems and their formulations.
X_{real}:
Input: A company produced two products and from each ...
Output: Maximize $100x + 300y$. Subject to: $50x + 100y < 1000$
$\mathcal{P}_{train}$
Train Data Generation
LLM ($\mathcal{M}_L$)
Train Data (X_{train})
SLM ($\mathcal{M}_S$)
$\mathcal{L}_{sce}(X_{train})$
OOD Sample Selection
Feedback Function $E(X_{val})$
OOD Samples (X_{fb})
$\mathcal{P}_{task} \oplus X_{real} \oplus X_{train} \oplus \mathcal{P}_{ood}$
$\mathcal{P}_{ood}$: Generate out-of-distribution samples that are significantly different given the training data.
Validation Data (X_{val})
LLM ($\mathcal{M}_L$)
$\mathcal{P}_{val}$
Validation Data Generation

Figure 2: Overview of the proposed data generation and knowledge distillation method, *GOLD*. $\oplus$: Concatenation.

onto the web. Further, the future-generation models considered in both works are not "task-specific" - they are trained with the goal of either meeting or exceeding the performance of the previous model, and the negative findings apply only in this case. Our goals are less ambitious, and tailored towards achieving the performance of the teacher model (i.e., an LLM) only on a *specific* task, rather than training a student model (i.e., an SLM) to reproduce the entire suite of capabilities as the LLM.

In addition, (Shumailov et al., 2023) indicates the poor generalizability of the distilled SLM due to the generation of only high likelihood data by the LLM. However, unlike the previous works, *GOLD* can effectively address this issue by iterative OOD-based feedbacks to the LLM to identify the failure modes of the SLM and improve its generalizability.

3 Method

Fig. 2 shows an overview of the proposed *GOLD*, a data generation and KD framework for language models. We iteratively transfer knowledge from an LLM, denoted by $\mathcal{M}_L$, to an SLM, denoted by $\mathcal{M}_S$, via the data generated by $\mathcal{M}_L$. The KD is performed for any task that falls within the realm of the expertise of $\mathcal{M}_L$. The framework operates with two inputs: a task definition $\mathcal{P}_{task}$ and a few samples of real data X_{real}.

In each iteration, *GOLD* generates a train set and an OOD validation set denoted by $(X_{\text{train}}, X_{\text{val}})$ that is respectively used to update the weights of $\mathcal{M}_S$ and to evaluate the updated $\mathcal{M}_S$. A feedback function, denoted by E, is then used to find the top OOD samples from X_{val} that are within a pre-defined upper and lower bound threshold. The sample selection procedure is performed regardless of the generated labels, which makes the feedback function robust to noisy labels. The selected samples, denoted by X_{fb}, are concatenated to the prompt for the next iteration of data generation. In the following, the details of the problem formulation, data generation procedure, the feedback function, and SLM training are discussed.

3.1 Problem Formulation

Our objective is to enhance the train data X_{train} to address failure modes of $\mathcal{M}_S$. This is achieved by automatically selecting the in-context examples present in the prompt $\mathcal{P}_{train}$ (Fig. 2) such that when $\mathcal{M}_S$ is trained on X_{train}, it exhibits optimal performance on X_{val}. X_{val} is generated to be *significantly different* from X_{train} in terms of topic and style in order to maximize the error of the SLM $\mathcal{M}_S$ on X_{val}. We argue that optimal knowledge transfer from $\mathcal{M}_L$ to $\mathcal{M}_S$ has been achieved when $\mathcal{M}_L$ only generates samples which $\mathcal{M}_S$ has already learned. To facilitate this, it is necessary to push $\mathcal{M}_L$ to generate less likely (i.e., away from the modes of its training distribution), yet accurate samples. It is important to note that LLMs often tend to generate high-likelihood samples, which can result in repeated or thematically similar samples.

We introduce a formulation that provides a conceptual framework for understanding our general approach and its underlying principles. Let denote the error of $\mathcal{M}_S$ on X_{val} as $\mathcal{E}\big(\mathcal{M}_S(X_{\text{val}}|X_{\text{train}})\big)$, the objective function can be written as:

$$\min_{X_{\text{train}}} \max_{X_{\text{val}}} \mathcal{E}\big(\mathcal{M}_S(X_{\text{val}}|X_{\text{train}})\big). \tag{1}$$

Unlike conventional optimization, the optimization of $\mathcal{E}$ leverages in-context learning where parameters of $\mathcal{M}_L$ are fixed. During each iteration, in-context examples in the prompt $\mathcal{P}_{\text{train}}$ guide X_{train} towards the failure modes of $\mathcal{M}_S$ (refer to Fig. 2). Similarly, X_{val} is steered towards more challenging samples using in-context examples in the prompt $\mathcal{P}_{\text{val}}$ (also see Fig. 2).

3.2 Data Generation

In the first iteration, given the prompt $\mathcal{P}_{train}$ including: 1) the task definition $\mathcal{P}_{\text{task}}$, and 2) n human-labeled data samples (with their labels), denoted by X_{real}, $\mathcal{M}_L$ generates a train batch of data:

$$X_{\text{train}}^{t_0} = \mathcal{M}_L(\mathcal{P}_{\text{task}} \oplus X_{\text{real}}), \tag{2}$$

where $\oplus$ denotes concatenation. $\mathcal{M}_L$ samples $X_{\text{train}}^{t_0}$ from the high likelihood tokens given the input prompt. Data generation with the aforementioned fixed prompt typically results in a distribution characterized by a high-density center. As indicated in prior works, such data generation procedure often leads to the disappearance of the tails in the distribution of human data observed by $\mathcal{M}_L$ (Shumailov et al., 2023). To address this issue, we initially propose the generation of an OOD validation set (with their labels).

At this stage, the OOD validation samples are defined in relation to $X_{\text{train}}^{t_0}$, which are considered as the in-distribution samples. Thus, the validation data at iteration t is generated by the prompt $\mathcal{P}_{\text{val}} = \mathcal{P}_{\text{task}} \oplus X_{\text{real}} \oplus X_{\text{train}}^{t} \oplus \mathcal{P}_{\text{ood}}$ as follows:

$$X_{\text{val}}^{t} = \mathcal{M}_L(\mathcal{P}_{\text{task}} \oplus X_{\text{real}} \oplus X_{\text{train}}^{t} \oplus \mathcal{P}_{\text{ood}}), \tag{3}$$

where $\mathcal{P}_{\text{ood}}$ prompts the generation of new data that significantly diverges from X_{train}^{t}, potentially in terms of topic, domain, or style. $\mathcal{M}_S$ is evaluated on X_{val}^{t} and based on the output logits, we identify failure samples (OOD samples) and form X_{fb}^{t}. Detailed selection procedure of X_{fb}^{t} is discussed in the next section.

The train data in the next iterations is generated by leveraging the feedback from the previous round. Given the prompt $\mathcal{P}_{train}$ consisting of task definition as well as real data and feedback samples, the train data at iteration $t+1$ is generated as follows:

$$X_{\text{train}}^{t+1} = \mathcal{M}_L(\mathcal{P}_{\text{task}} \oplus X_{\text{real}} \oplus X_{\text{fb}}^{t}). \tag{4}$$

Ideally, the train samples X_{train}^{t+1} should be accurate and closely resemble the provided feedback samples X_{fb}^{t}. By updating the weights of the SLM $\mathcal{M}_S$ with these newly generated samples X_{train}^{t+1}, we can address the failure modes of $\mathcal{M}_S$ encountered in the previous iteration and enhance its generalizability. In other words, the process of validation and train data generation acts as two competing agents. Their goal is to improve the generalizability of the SLM and to challenge it with a progressively harder validation set.

3.3 OOD-Based Feedback

GOLD employs an energy function to detect OOD samples, which are then used as a feedback to the LLM. Energy functions have shown to be effective in identifying OOD samples in classification and regression tasks (Liu et al., 2020; Akbari et al., 2021; Gholami et al., 2023). The concept of energy-based models (EBM) is pivotal in this context.

Considering $\mathcal{M}_S$ as a classification model with C classes, we can compute the free energy score corresponding to an input sample x as follows:

$$E(x) = -\log \sum_{c}^{C} e^{\mathcal{M}_S^{(c)}(x)}, \tag{5}$$

where $\mathcal{M}_S^{(c)}(x)$ is the output logit of c-th class.

The above-mentioned procedure can also be extended to sequence-to-sequence models (Akbari et al., 2022), where a sequence of tokens is generated as output. Given $\mathcal{M}_S$ as our model, for each token, $\mathcal{M}_S$ predicts a class from a dictionary consisting of K vocabularies. Consider an input-output $(x, \hat{y}, l)$, where x is the input text, $\hat{y}$ is the predicted sequence text, and l is the corresponding logits of $\hat{y}$, denoted as $l : [l_i]_{i=1}^{L}$ where L is the length of the returned sequence and $l_i \in \mathbb{R}^K$. Following Eq. 5, we calculate the energy score of the input sequence x as:

$$E_s(x) = -\frac{1}{L} \sum_{i=1}^{L} \log \sum_{k=1}^{K} e^{l_i^k}, \tag{6}$$

that returns the average energy score over all tokens in the predicted sequence. The energy score calculation is performed in an unsupervised manner, where the data labels are not required. Thus, we expect the energy score to be robust to potential noisy labels generated by the LLM. We then select the samples with low negative energy scores from X_{val} as follows:

$$X_{\text{fb}} = \{X_{\text{val}}^{i} \mid \alpha < -E_s(X_{\text{val}}^{i}) < \beta,\ X_{\text{val}}^{i} \in X_{\text{val}}\}, \tag{7}$$

where β is the upper threshold to select samples with low negative energy score (OOD samples) and α is a lower threshold to exclude samples that are very noisy and drastically OOD. The selected OOD samples (X_{fb}) along with their labels are then incorporated in the prompt $\mathcal{P}_{train}$ for generating train data in the next iteration.

3.4 SLM Training

In each iteration, the generated train data X_{train}^{t} is used to train and update the SLM. Since there

Task	Method	Prompt
ANLI	***GOLD***, ZeroGen, ProGen, P2Model	[Below are 3 examples of an natural language inference dataset. Samples include a “hypothesis” and a “premise”. The label of the sample is: 1. “entailment” if the premise entails the hypothesis, 2. “neutral’ if the premise neither entails nor contradicts the hypothesis. 3. “contradiction” if premise contradicts hypothesis.]$^{\mathcal{P}1}$ [Sample 1, Sample 2, Sample 3]$^{\mathcal{P}2}$ [Generate a novel sample of data.]
	Few-shot	[...]$^{\mathcal{P}1}$[...]$^{\mathcal{P}2}$[What is the label of the below sample?] [some sample]

Table 1: Example prompts used for data generation. *Few-shot*: inference with the LLM given a few data samples.

might be noisy data labels inaccurately generated by the LLM, we use symmetric cross-entropy loss (Wang et al., 2019b) for training the SLM in our work to address the potential noises in the samples generated by LLM. The symmetric cross-entropy loss consists of two terms including reverse cross-entropy and cross-entropy as follows:

$$\begin{aligned}\mathcal{L}_{sce} = \frac{-1}{N}\sum_{i=1}^{N}\big(\lambda\sum_{k=1}^{K}\hat{y}_k\log(y_k) \\ +\sigma\sum_{k=1}^{K} y_{i,k}\log(\hat{y}_{i,k})\big)\end{aligned} \tag{8}$$

where $\hat{y}$ and y are respectively the predicted and ground-truth labels (generated as text). K is the vocabulary size of the tokenizer of $\mathcal{M}_S$, N is the total number of samples, and λ and σ are hyperparameters that adjust the weight of reverse cross entropy and cross-entropy. We set λ=1.0 and σ=0.1, which are obtained experimentally.

4 Experiments

In this section, the performance of *GOLD* is analyzed and compared with previous works over different classification and sequence-to-sequence tasks. First, we describe the experimental settings including the datasets (tasks), model architectures, prompts, and baselines. Following that, the experimental results are quantitatively and qualitatively discussed and compared with the baselines. An extensive set of ablations over *GOLD* components, size of LLM and SLM, and the number of generated samples is also provided.

4.1 Settings

Tasks. We use 6 different classification tasks including ANLI, MNLI, QNLI, WNLI, RTE, and MRPC (Wang et al., 2019a), where accuracy is used as the evaluation metric. We further use 3 sequence-to-sequence tasks including SQUAD (Rajpurkar et al., 2016), Adversarial QA (Adv-QA) (Bartolo et al., 2020), and SVAMP (Patel et al., 2021). SQUAD and Adv-QA are question and answer (QA) datasets while SVAMP is a math word problem. We respectively use Exact Match (EM) and accuracy as the evaluation metrics for QA and SVAMP datasets. In order to further evaluate the performance of *GOLD* on less explored novel tasks, we perform experiments on recently released dataset NL4OPT (Ramamonjison et al., 2023). Input and output of NL4OPT dataset are respectively description of an optimization problem and its corresponding optimization formulation. We use ROUGE-L (Lin, 2004) as the evaluation metric for this task. The validation sets from the above-mentioned datasets are used for the numerical analysis of all the methods in this paper.

LLM and SLM. We respectively use LLaMA2-7B (Touvron et al., 2023) and pre-trained T5-base (220M) (Raffel et al., 2020) as the LLM and SLM for all our experiments. As T5 is a sequence-to-sequence model, we define all of the tasks including the classification ones as sequence-to-sequence tasks. Doing so, our framework is flexible to be applied to various NLP tasks. Details of the datasets used for pre-training the T5 model used in this paper are provided in (Raffel et al., 2020).

Initial Prompts. For all the experiments in this paper, we use a fixed, general prompt including a task definition and 3 samples of real data. Table 1 shows an example prompt for the ANLI task used by *GOLD* and the other methods. We use the same prompt for different methods for fair comparison. The prompts for other tasks are given in Appendix D.

Baselines. We compare our results with three prior arts including ZeroGen (Ye et al., 2022a), ProGen (Ye et al., 2022b), and Prompt2Model (Viswanathan et al., 2023). Although ZeroGen and ProGen are initially designed for classification tasks, we modify their implementation to be further applicable to sequence-to-sequence tasks. We also compare our results with the pre-trained SLM, fine-tuned SLM, and few-shot performance of the specified LLM (i.e., LLaMA2-7B). The pre-trained SLM is indeed the model that has been simultaneously pre-trained on a variety of datasets including all the downstream ones (except ANLI and WNLI).

Model	Model Size	Data	Method	ANLI	MNLI	QNLI	WNLI	RTE	MRPC	Ave.
T5-base	220M	Full	Fine-Tuned	43.1	86.6	93.7	78.8	80.1	87.5	78.3
			Pre-Trained	29.0	56.6	88.3	52.1	68.5	75.0	61.6
LLaMA2	7B	-	Few-shot	36.0	41.5	55.3	53.5	62.4	65.4	52.3
LLaMA2 → T5-base	220M	3K	P2Model	34.4	59.5	62.2	56.3	58.8	75.0	57.7
			ZeroGen	34.6	56.1	88.5	54.9	62.1	84.3	63.4
			ProGen	34.3	55.1	85.9	57.7	66.0	80.3	63.2
			GOLD	**35.7**	**62.5**	**91.7**	**57.7**	**69.6**	**85.0**	**67.1**

Table 2: Comparison results on classification tasks in terms of accuracy. We use T5-base (220M) as SLM and LLaMa2 (7B) as LLM. *Pre-Trained*: simultaneously pre-trained on a variety of datasets including all the downstream ones (except ANLI and WNLI). *Fine-Tuned*: specifically fine-tuned over the downstream dataset. *Few-shot*: inference with the LLM given a few data samples.

Model	Model Size	Data	Method	SQUAD	Adv-QA	SVAMP	NL4OPT
				(EM%)	(EM%)	(Acc%)	(ROUGE-L)
T5-base	220M	Full	Fine-Tuned	77.3	38.5	53.7	89.2
			Pre-Trained	74.7	25.3	-	-
LLaMA2	7B	-	Few-shot	15.2*	14.6*	27.3	54.4
LLaMA2 → T5-base	220M	3K	ZeroGen	69.4	21.3	20.0	67.9
			ProGen	68.1	20.5	23.7	68.5
			P2Model	74.4	25.0	**26.0**	71.5
			GOLD	**75.2**	**25.5**	25.3	**72.8**

Table 3: Comparison results on seq-to-seq tasks. *EM*: Exact Match. **Note**: pre-trained T5-base has not seen SVAMP & NL4OPT tasks (nor any similar dataset). *: obtained with the general prompts in Table 1. EM of 54.7 & 25.1 is respectively achieved for SQUAD and Adv-QA in case of using specific prompts in (Touvron et al., 2023).

On the other hand, the fine-tuned version is the model specifically fine-tuned over a downstream dataset. Moreover, the few-shot results of the LLM is achieved by doing the inference with the LLM given a few data samples. Unless otherwise mentioned, for the experiments corresponding to all the tasks and methods in this paper, we generated 3K samples using the LLM in 375 iterations.

4.2 Results

Quantitative Results. Table 2 shows the results of our method on the classification tasks. The results are compared with the prior arts, the LLM, and the pre-trained/fine-tuned SLMs. It is shown that our method (with an average accuracy of 67.1%) improves the performance of the pre-trained SLM (with an average accuracy of 61.6%) over all the six tasks. Our method demonstrates a 4% improvement over ZeroGen and ProGen, and also 10% improvement over P2Model. Compared to the LLM's few-shot with an average accuracy of 52.3%, we substantially obtain better results with 14% margin. Despite the LLM's poor performance over the validation sets of the tasks, the data generated by the LLM is sufficiently accurate to facilitate knowledge transfer to the SLM. To study this, we used GPT-4 (OpenAI, 2023) to generate the ground truth labels of the generated data, and then evaluated the LLM's accuracy. To this end, a significant accuracy of 80.3% and 72.1% was respectively observed over the data generated for QNLI and RTE.

The comparison results for the sequence-to-sequence tasks are given in Table 3, where *GOLD* is better than prior arts, few-shot results of the LLM, and pre-trained SLM on SQUAD and Adv-QA datasets. Note that the evaluation metric of SQUAD and Adv-QA datasets is Exact Match (EM) that requires the output to be exactly similar to the labels. Moreover, as mentioned in Section 4.1, we used a consistent prompt format in all of our experiments including the LLM's (i.e., LLaMA2) few-shot results for simplicity and generality. Therefore, LLaMA2 provides lower EM accuracy (i.e., 15.2 and 14.6 for SQUAD and Adv-QA) compared to the results reported in their original paper (Touvron et al., 2023). However, if we use the specific prompts from LLaMA2's paper, the results are improved to 54.7 and 25.1 for SQUAD and Adv-QA, respectively.

Note that the SVAMP dataset was not included in the train set of the pre-trained SLM. However, the LLM may have been exposed to similar datasets during its training phase. *GOLD* (with an accuracy of 25.3%) is the second best on SVAMP dataset

Method	ID	Random Samples
ZeroGen	1	**Sentence1:** The new smartphone from Apple has a cutting-edge AI assistant that can learn and adapt to the user's preferences. , **Sentence2:** The AI assistant from Samsung is more advanced and can perform complex tasks with greater ease. , **Label:** entailment
	2	**Sentence1:** The new smartphone features a cutting-edge AI-powered camera that can automatically detect and enhance low-light photos., **Sentence2:** The latest smartphone has a revolutionary camera that can take high-quality photos in any lighting conditions., **Label:** neutral
ProGen	1	**Sentence1:** The new smartphone has a powerful processor that enables it to run complex apps with ease., **Sentence2:** The new smartphone has a sleek design that makes it easy to hold and use., **Label:** neutral
	2	**Sentence1:** The new smartphone from Apple is revolutionizing the way people communicate. , **Sentence2:** The new smartphone from Samsung is a total failure and will not last more than a year., **Label:**contradiction
P2Model	1	**Sentence1:** The company's new marketing strategy aims to increase sales by targeting a younger demographic., **Sentence2:** The company will see an increase in sales as a result of the new marketing strategy., **Label:** entailment
	2	**Sentence1:** The new AI model can generate realistic images of objects and scenes based on text descriptions. , **Sentence2:** The model can create images of objects and scenes that are indistinguishable from real-world examples., **Label:** entailment
GOLD	1	**Sentence1:** The 2022 Winter Olympics are scheduled to take place in Beijing, China from February 4 to 20, 2022., **Sentence2:** The 2022 Winter Olympics are expected to be a showcase of the latest technology and innovation in the sporting world. , **Label:** neutral
	2	**Sentence1:** The new electric car model is environmentally friendly and reduces carbon emissions., **Sentence2:** The new electric car model is powered by fossil fuels and emits more carbon than a traditional gasoline-powered car., **Label:** contradiction

Table 4: Two data samples generated by ZeroGen, ProGen, and *GOLD* randomly sampled from the generated dataset for ANLI task. *GOLD* generates more diverse samples compared to baseline methods.

Method	Model	ANLI	MNLI	QNLI	WNLI	RTE	MRPC	SVAMP	NL4OPT	Ave.
Pre-Trained	T5-small (60M)	28.7	47.4	86.3	43.6	52.7	69.3	-	-	54.7
	T5-base (220M)	29.0	56.6	88.3	52.1	68.5	75.0	-	-	61.6
	T5-large (770M)	32.6	58.9	90.1	45.0	72.5	72.3	-	-	61.9
GOLD	LLaMA2 →T5-small	32.6	56.1	83.5	46.4	63.5	75.9	13.3	70.4	55.2
	LLaMA2 →T5-base	35.7	62.5	91.7	**56.3**	69.6	**85.0**	25.3	**72.8**	62.4
	LLaMA2 →T5-large	**35.7**	**63.3**	**93.3**	52.1	**74.3**	83.1	**29.7**	72.5	**63.0**

Table 5: The performance of *GOLD* with different SLM sizes. The bigger the model, the better the performance (except WNLI and MRPC that follow the same trend as in the pre-trained SLMs).

after P2Model (with an accuracy of 26.0%).

Table 3 also presents the results on NL4OPT, a recently released dataset. The SLM has not encountered this dataset or a similar task before. Likewise, the LLM probably has not been exposed to this dataset or this specific task, although it might have encountered optimization problems from other sources. We aim to explore the potential of *GOLD* in preparing SLMs for new tasks that the LLM has not directly encountered before. Our proposed method outperforms prior works by obtaining a ROUGE-L of 72.8%. More results on other seq-to-seq tasks are given in Appendix B.

As expected, the best performance is achieved by the SLMs specifically fine-tuned on the train set of each downstream dataset in a supervised way, i.e., the upper-bound results in this work.

Qualitative Results. Table 4 shows two samples that were randomly selected from the datasets generated by ZeroGen, ProGen, P2Model, and *GOLD*. The samples generated by ProGen and ZeroGen are closely related, while those produced by P2Model and *GOLD* exhibit greater diversity. More results are given in Appendix D.

4.3 Ablations

SLM Size. Table 5 shows the experimental results with different SLM models including T5-small (60M), T5-base (220M), and T5-large (770M). In general, as the SLM size increases, better results are obtained by both pre-trained and distilled versions. However, there are two exceptions including WNLI and MRPC for which T5-base outperforms T5-large. This is due to the overfitting of the larger model on the small datasets of WNLI and MRPC.

Components of *GOLD*. Table 6 summarizes the ablation on the main components of *GOLD*. Ablating the feedback function (i.e., V3 in the table) decreases the accuracy on RTE and MNLI datasets by 6% and 3%, respectively. Similarly, ablating the noise-robust SCE loss (i.e., V2) decreases the accuracy by about 3% on RTE and MNLI datasets. This

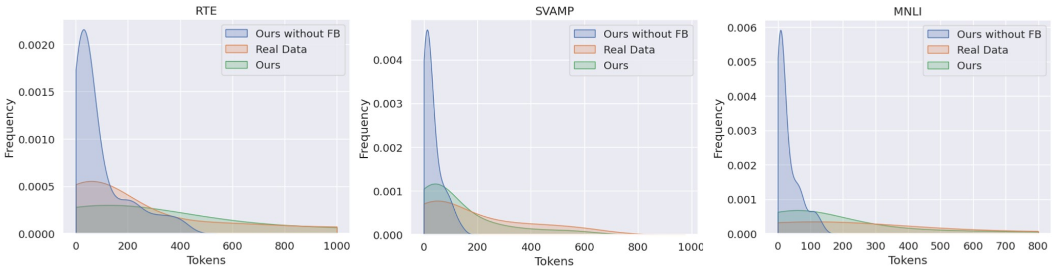

Figure 3: The distribution of generated data by our method with and without OOD-based feedback.

	KD	FB	SCE	MNLI	QNLI	SVAMP	RTE
V0				56.6	88.3	-	68.5
V1	✓			55.4	90.6	20.0	67.1
V2	✓	✓		59.5	91.4	23.3	68.6
V3	✓		✓	59.1	91.0	17.3	63.1
GOLD	✓	✓	✓	**62.5**	**91.7**	**25.3**	**69.6**

Table 6: Ablation on the main components of *GOLD*. *V0*: pre-trained SLMs. *KD*: knowledge distillation by vanilla data generation. *FB*: the feedback function in the course of data generation. *SCE*: symmetric cross entropy used for training the distilled SLM.

ablation study shows that having noise-robust loss is critical when fine-tuning SLMs using synthetic generated data. Moreover, the feedback function is effective to boost the performance.

Distribution of Generated Data. Figure 3 illustrates the distribution of the data generated by *GOLD* with and without OOD-based feedback compared with that of real data. The distribution plots for example classification and sequence-to-sequence tasks are shown. As noted in the plots, the data generated using our proposed OOD-based feedback mechanism provides a more long-tailed distribution (i.e., including low-probability events) that is closer to the real data distribution. In contrast, vanilla data generation results in only high-likelihood data, while disappearing low probability tokens. More plots are given in Appendix C.

Method	Model	MNLI	WNLI	RTE
Few-shot	LLaMA2-7B	41.5	53.5	62.4
	LLaMA2-70B	55.6	63.4	74.7
GOLD	LLaMA2-7B → T5-base	60.0	57.7	69.3
	LLaMA2-70B→ T5-base	**62.4**	**65.7**	**71.1**

Table 7: Ablation on LLM size with 1K samples.

LLM Size. Table 7 presents an ablation study conducted on the size of the LLM. We carried out the experiments with only 1K samples using two different versions of LLM: LLaMA-70B and LLaMA-7B. The results indicate that the few-shot performance of the 70B LLM surpasses that of the 7B one. Furthermore, when the SLM is distilled from the 70B LLM, it demonstrates superior performance compared to the SLM distilled from the 7B LLM. This suggests that our framework can potentially enhance the results when a larger LLM is employed. However, it is worth noting that it becomes increasingly challenging to exceed the few-shot results of larger LLMs. For instance, as shown in Table 7, the distilled SLM consistently outperforms the few-shot performance of LLM when the 7B version is used. However, this trend does not hold when the 70B LLM is utilized.

Dataset Size. Table 8 presents the performance of our method when the volume of generated data varies from 0 to 5K samples. As observed, increasing the number of samples (that are mostly OOD) does not necessarily improve the overall performance of *GOLD*. However, this is only the case for the tasks on which the SLM has been pre-trained using real data. In other words, our pre-trained SLMs have already seen enough high likelihood samples, which are supposed to be the majority compared to the OOD samples. Thus, there is no need for big amount of OOD samples and a small, but robust set of samples is sufficient to improve the generalizability of the distilled SLM. On the other hand, for the SVAMP that the pre-trained SLM has not encountered before, increasing the volume of data consistently enhances the accuracy.

#Samples	MNLI	SVAMP	QNLI	RTE
0	56.6	0	88.3	68.5
1000	60.0	17.3	91.5	69.3
2000	**63.4**	22.7	90.4	69.3
3000	62.5	25.3	**91.7**	**69.6**
4000	62.5	25.3	90.9	68.9
5000	62.9	**28.3**	89.4	69.3

Table 8: Ablation on the number of generated samples.

4.4 Hyperparameters Selection

α and β in Eq. 7. We utilize a lower and an upper threshold, denoted as α and β, respectively, in Eq. 7 for the selection of OOD samples. For each batch of data, we set these thresholds to select samples that fall within the range of 50% to 80% of the lowest negative energy scores. As shown in (Shumailov et al., 2023), most of the samples generated by LLMs are high-likelihood samples. Therefore, specifying the upper threshold as 50% guarantees that all of the excluded samples are high likelihood. Indeed, extensive hyperparameter tuning on each downstream dataset improves the results. However, for simplicity and generality, we used a fixed set of hyperparameters on all tasks.

σ and λ in Eq. 8. As we do not use any validation set to select the hyperparameters, the upper and lower thresholds in Eq. 8 were chosen based on (Wang et al., 2019b), where σ=0.1 and λ=1 in most of the experiments. (Wang et al., 2019b) shows that in symmetric cross-entropy loss, large σ (e.g., between 0.5 and 1.0) tends to cause more overfitting, while small σ (e.g., between 0.01 and 0.1) can help ease the overfitting of CE. We also experimented on different values of σ, while fixing λ to 1 over the ANLI dataset. As shown in Table 9, σ=0.1 provides the best performance. Overall, the value σ does not have a significant effect on our method's performance and we still outperform prior arts with different values.

σ	0.01	0.1	1
Accuracy	0.354	**0.357**	0.350

Table 9: Analysis of σ and λ values in Eq. 8.

4.5 Running Time Analysis

Table 10 presents the running time, in terms of seconds per sample, of *GOLD* and prior works. The analysis was done on 4 NVIDIA Titan Xp GPUs. As seen, ProGen exhibits the highest running time, which is due to using an expensive influence function to measure the quality of generated samples. On the other hand, ZeroGen and Prompt2Model, which lack any feedback and evaluation function and do not append feedback samples to the prompt, are slightly faster than *GOLD*.

Please note that the time cost of *GOLD* and the previous works linearly increases with the number of samples. Consequently, the total time required to generate 3K samples for each task is approximately 4 hours for ZeroGen, around 4.5 hours for both *GOLD* and Prompt2Model and about 6 hours for ProGen. It should also be noted that we only append 2-3 OOD samples from the previous iteration to generate a new batch of data, which only adds an overhead of 0.4 seconds per sample (20min for 3k samples).

Method	ZeroGen	ProGen	P2Model	*GOLD*
Time (s)	4.9	7.3	5.3	5.4

Table 10: Running time analysis.

4.6 Lexical Diversity

Table 11 shows the lexical diversity of our generated data compared to prior arts. Lexical diversity is defined as the number of unique words divided by the total number of words in the dataset. P2Model generates the most diverse samples of data due to the high temperature value, and *GOLD* is the second best in terms of the diversity of data. P2Model increases the temperature of the LLM during data generation to enhance diversity. However, we argue that relying on temperature can potentially compromise the accuracy of the generated samples. To this end, we utilized GPT-4 to evaluate the accuracy of the data generated by the LLM for P2Model with a high temperature, where an accuracy of 38.6% and 59% was respectively obtained on the QNLI and RTE. For our method, where the LLM temperature is fixed to the default value of 1, we achieved an accuracy of 80.3% and 72.1% for the same tasks.

	Lexical Diversity			
Method	MNLI	QNLI	MRPC	SQUAD
ZeroGen	1.5	3.7	2.7	1.3
ProGen	1.5	3.7	2.8	1.9
P2Model	**7.6**	10.2	**7.0**	**10.0**
GOLD	<u>6.4</u>	**11.5**	<u>6.6</u>	<u>3.1</u>

Table 11: Lexical diversity of the generated data by *GOLD* and prior works. P2Model uses high temperature for more diversity.

5 Conclusion

In this paper, we proposed a task-agnostic framework for data generation and KD from LLMs to SLMs that is applicable to any NLP task, including new tasks. In order to improve the generalizability of the distilled SLMs, we introduced an iterative OOD-empowered feedback mechanism. An energy-based OOD evaluation approach was also proposed to handle noisy data generated by LLMs. With an extensive set of experiments, we showed that our method achieved state-of-the-art results on a variety of classification and sequence-to-sequence downstream tasks in NLP. Future direction of this work includes applying the same technique for other data modalities such as images.

6 Limitation

Despite its promising results, *GOLD* does have certain limitations that warrant further investigation. One of the primary challenges we encountered was identifying the failure modes of the SLM and subsequently guiding the data generation procedure. This approach is susceptible to generating samples that are drastically different, which can negatively impact the performance of the distilled SLM. To mitigate this issue, we implemented an upper-bound threshold for selecting OOD samples. However, our observations indicate that some data points in the feedback loop still deviate from the correct format of real data.

Future work should focus on incorporating a data valuation method. This would enable the automatic evaluation of selected samples before they are fed back into the LLM. Such approach could significantly enhance the reliability and accuracy of the data generation process, leading to improved performance of the distilled models.

In this study, we demonstrated the potential of preparing SLMs for NL4OPT, a novel task for which the LLM has not been specifically trained. This approach opens up new avenues for leveraging language models in diverse applications. However, the performance of our proposed method on such innovative tasks requires further investigation. Future research should focus on evaluating the effectiveness of *GOLD* across a broader range of user-defined tasks.

References

Asma Ben Abacha, Wen-wai Yim, Yadan Fan, and Thomas Lin. 2023. An empirical study of clinical note generation from doctor-patient encounters. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2291–2302.

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. 2023. Generalized knowledge distillation for auto-regressive language models.

Mohammad Akbari, Amin Banitalebi-Dehkordi, and Yong Zhang. 2021. EBJR: Energy-based joint reasoning for adaptive inference. *BMVC 2021*.

Mohammad Akbari, Amin Banitalebi-Dehkordi, and Yong Zhang. 2022. E-LANG: Energy-based joint inferencing of super and swift language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5229–5244, Dublin, Ireland. Association for Computational Linguistics.

Mohammad Akbari, Amin BANITALEBI DEHKORDI, Tianxi Xu, and Yong Zhang. 2023. System and method for unsupervised multi-model joint reasoning. US Patent App. 18/074,915.

Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the ai: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Mohsen Gholami, Mohammad Akbari, Xinglu Wang, Behnam Kamranian, and Yong Zhang. 2023. ETran: Energy-based transferability estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18613–18622.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2023. Knowledge distillation of large language models.

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. The false promise of imitating proprietary llms.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8003–8017, Toronto, Canada. Association for Computational Linguistics.

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

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. 2020. Energy-based out-of-distribution detection.

OpenAI. 2023. GPT-4 Technical Report. *arXiv e-prints*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094, Online. Association for Computational Linguistics.

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.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Rindranirina Ramamonjison, Timothy T. Yu, Raymond Li, Haley Li, Giuseppe Carenini, Bissan Ghaddar, Shiqi He, Mahdi Mostajabdaveh, Amin Banitalebi-Dehkordi, Zirui Zhou, and Yong Zhang. 2023. Nl4opt competition: Formulating optimization problems based on their natural language descriptions.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross Anderson. 2023. The curse of recursion: Training on generated data makes models forget.

Nassim Nicholas Taleb. 2007. Black swans and the domains of statistics. *The American Statistician*, 61(3):198–200.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Vijay Viswanathan, Chenyang Zhao, Amanda Bertsch, Tongshuang Wu, and Graham Neubig. 2023. Prompt2model: Generating deployable models from natural language instructions.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019a. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019*.

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. 2019b. Symmetric cross entropy for robust learning with noisy labels. In *IEEE International Conference on Computer Vision*.

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.

A Appendix Introduction

In this appendix, we provide 1) more quantitative results on text generation tasks, 2) more distribution plots of the generated data, 3) the specific prompts utilized for each task, and 4) further qualitative results.

To ensure the reproducibility of our experiments, we have also shared the relevant code with detailed instructions, which is available here[2].

B Results on Text Generation Tasks

We have conducted experiments on text-to-text tasks, namely NL4OPT and SVAMP in the main body of the paper. In this section, we conduct another set of experiments on a recently released, challenging text generation task, known as Medical-Dialogue-to-Note (Abacha et al., 2023). The objective of this task is to generate a note that a doctor would take during a patient visit, given a dialogue between the patient and the doctor. This dataset has been recently released, and we assume that both the LLM and SLM have not been exposed to this dataset. As shown in Table 12, the SLM (i.e., T5-base) distilled using *GOLD* achieves a ROUGE-L score of 0.198, which is twice better than the pre-trained T5 with 0.101 and is comparable with the LLM (i.e., LLaMA2-7B) few-shot results with 0.218.

	PT-SLM	FT-SLM	LLM	*GOLD*
ROUGE-L	0.101	0.329	0.218	0.198

Table 12: Results on Medical-Dialogue-to-Note text generation task. PT: pre-trained; FT: fine-tuned.

C Generated Data Distribution

Figure 4 illustrates the distribution plots of the data generated by *GOLD* with and without OOD-based feedback compared with that of real data. The distribution plots for sequence-to-sequence tasks including SVAMP, SQUAD, and Adv-QA are shown.

D Prompts and Qualitative Results

Tables 13, 14, and 15 provide the detailed prompts that were used to generate data for each task. We provide three real samples from the train set of each task in the prompt. When we have OOD samples as the feedback, they will be added after the real samples. Also, Table 16 provides samples of data generated for each of the tasks.

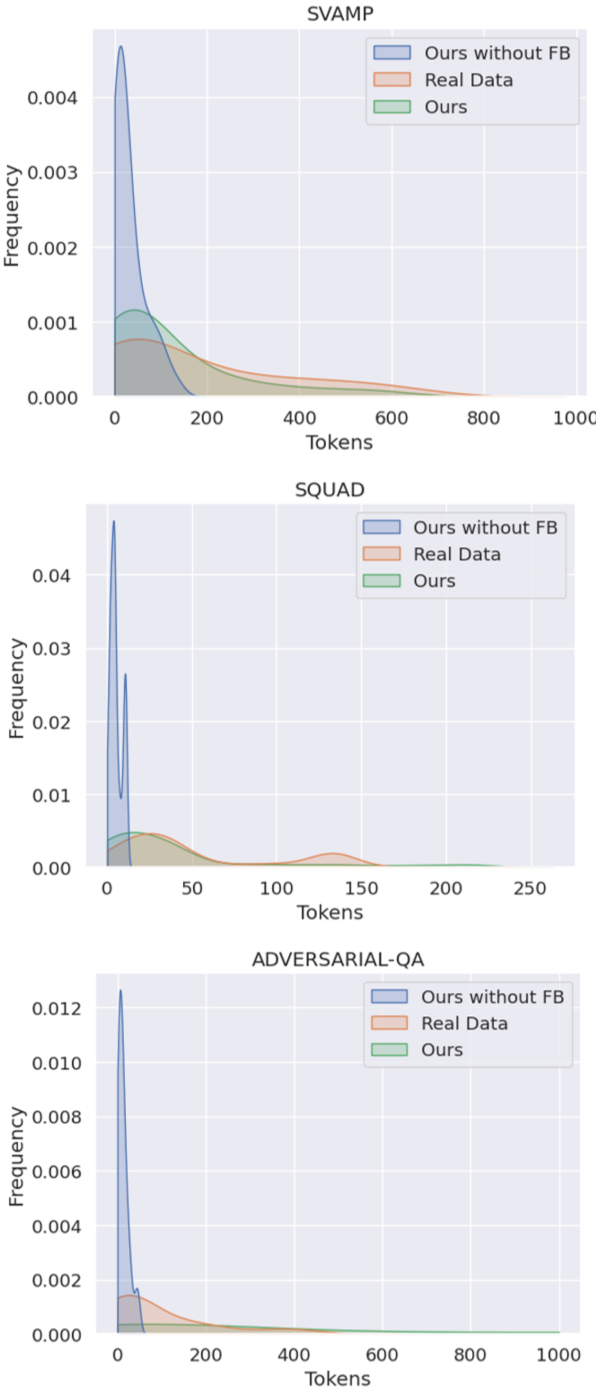

Figure 4: The distribution of generated data by our method with and without OOD feedback.

[2] https://developer.huaweicloud.com/develop/aigallery/notebook/detail?id=9d770d1f-3758-4d0f-99d4-3346abbe1546

Data	Method	Prompt
NL4OPT	*GOLD*, ZeroGen, ProGen, P2Model	The above are examples of NL4OPT dataseta and below are three samples of 'NL4OPT' dataset. The samples have a 'problem' which is a linear optimization problem and a 'formulation' which is the formulation of the optimization problem. 1. problem: A hotel employs cleaners and receptionists. Cleaners earn \$500 per week and receptionists earn \$350 per week. The hotel requires a minimum of 100 workers of whom at least 20 must be receptionists. To keep the hotel clean and running smoothly, the number of receptionists should be at least a third of the number of cleaners. The hotel wants to keep the weekly wage bill below \$30000. Formulate a LP to minimize the wage bill. formulation: Variables: x: sled dogs, y: trucks, Objective Function: maximize 100 x + 300 y ,Constraints:50 x + 100 y $\leq$ 1000, x $\leq$ y 2. problem: An office supply company makes two types of printers: color printers and black and white printers. Different sections of the factory with different teams produce each printer. The color printer team can produce at most 20 color printers per day while the black and white printer team can produce at most 30 black and white printers per day. Both teams require use of the same paper tray installing machine and this machine can make at most 35 printers of either type each day. Color printers generate a profit of \$200 per printer while black and white printers generate a profit of \$70 per printer. How many of each printer should be made to maximize the company's profit? formulation: Variables: x: color printers, y: black and white printers, Objective Function: maximize 200 x + 70 y, Constraints: x $\leq$ 20, y $\leq$ 30, x + y $\leq$ 35 3. problem: An accounting firm has senior accountants earning \$3000 per week and junior accountants earning \$1000 per week. The contracts with companies to provide accounting services require at least 100 accountants, of whom at least 5 must be senior accountants. To make sure there is enough experience on the accounting team, the number of senior accountants should be at least a third of the number to junior accountants. The firm wants to keep the weekly wage bill below \$150000. Formulate an LP to minimize the wage bill. formulation: Variables: x: senior accountants, y: junior accountants, Objective Function:minimize 3000 x + 1000 y, Constraints: x + y $\geq$ 100, x $\geq$ 51, x $\geq$ 0.33 y, 3000 x + 1000, y $\leq$ 150000. The above are samples of NL4OPT data. Think step by step and give me a novel sample of NL4OPT dataset.

Table 13: Sample prompts used for NL4OPT dataset. We use the same prompt for different methods for fair comaprison.

Data	Method	Prompt
QNLI	*GOLD*, ZeroGen, ProGen, P2Model	Below are three samples of QNLI dataset. Samples include a Question and a Sentence. The label of sample is 1.’entailment’ if the answer of the Question is in the Sentence and 2.’not_entailment’ if the answer of the Question is not in the Sentence. 1. Sentence: He must do this by collecting the multiple Tears of Light; once all the Tears of Light are collected for one area, he restores that area’s Light Spirit. Question: What does Link have to gather in order to complete each area? Label: entailment 2. Sentence:Prior to this time congressional parties were often relatively disorganized, so it was not always evident who functioned as the opposition floor leader. Question: Why was minority leader position created?. Label: entailment 3. Sentence:This view is shared by other researchers who argue that the ancestors of the American Indians were the first to separate from the great Asian population in the Middle Paleolithic. Question:Who have studies of the mtDNA of Turkic-speaking peoples shown they’re closest to genetically? Label: not_entailment The above are three samples of QNLI data. Think step by step and give me a novel sample of QNLI data with <not_entailment> label.
RTE-WNLI	*GOLD*, ZeroGen, ProGen, P2Model	You are a helpful ’Assistant’. You only reply once as ’Assistant’. Do not pretend to be a ’User’. Below are three samples of RTE dataset. Samples include a ’Sentence1’ and a ’Sentence2’. The label of sample is ’entailment’ if the answer of the ’Sentence1’ entails ’Sentence2’ and ’not_entailment’ if ’Sentence1’ does not entail ’Sentence2’. 1. Sentence1: No Weapons of Mass Destruction Found in Iraq Yet. Sentence2: Weapons of Mass Destruction Found in Iraq. Label: not_entailment 2. Sentence1: A place of sorrow, after Pope John Paul II died, became a place of celebration, as Roman Catholic faithful gathered in downtown Chicago to mark the installation of new Pope Benedict XVI. Sentence2: Pope Benedict XVI is the new leader of the Roman Catholic Church. Label: entailment 3. Sentence1:Herceptin was already approved to treat the sickest breast cancer patients, and the company said, Monday, it will discuss with federal regulators the possibility of prescribing the drug for more breast cancer patients. Sentence2:Herceptin can be used to treat breast cancer. Label: entailment The above are three samples of RTE data. Think step by step and give me a novel sample of RTE data with <entailment> label.
MRPC	*GOLD*, ZeroGen, ProGen, P2Model	Below are three samples of MRPC dataset. Samples include a ’Sentence 1’ and a ’Sentence 2’. The label of sample is ’equivalent’ if the ’Sentence 1’ and the ’Sentence 2’ are paraphrases of each other. The label is ’not_equivalent’ if ’Sentence 1’ and ’Sentence 2’ are not semantically equivalent. 1. Sentence 1: Amrozi accused his brother , whom he called the witness , of deliberately distorting his evidence . Sentence 2: Referring to him as only the witness , Amrozi accused his brother of deliberately distorting his evidence . Label: equivalent 2. Sentence 1: Yucaipa owned Dominick ’s before selling the chain to Safeway in 1998 for $ 2.5 billion . Sentence 2: Yucaipa bought Dominick ’s in 1995 for $ 693 million and sold it to Safeway for $ 1.8 billion in 1998 . Label: not_equivalent 3. Sentence 1: Around 0335 GMT , Tab shares were up 19 cents , or 4.4 % , at A $ 4.56 , having earlier set a record high of A $ 4.57 . Sentence 2: Tab shares jumped 20 cents , or 4.6 % , to set a record closing high at A $ 4.57 . Label: not_equivalent The above are three samples of MRPC data. Think step by step and give me a novel sample of RTE data with <equivalent> label.
SVAMP	*GOLD*, ZeroGen, ProGen, P2Model	Below are 3 examples of SVAMP dataset. Samples include a ’Body’ which explains a simple math problem and a ’Quesiton’ which ask a question from the ’Body’. 1. Body:There are 87 oranges and 290 bananas in Philip’s collection. If the bananas are organized into 2 groups and oranges are organized into 93 groups. Question: How big is each group of bananas? Equation:(290.0 / 2.0) 2. Body: Marco and his dad went strawberry picking. Marco’s dad’s strawberries weighed 11 pounds. If together their strawberries weighed 30 pounds. Question: How much did Marco’s strawberries weigh? Equation: (30.0 - 11.0) 3. Body: Edward spent $ 6 to buy 2 books each book costing him the same amount of money. Now he has $ 12. Question: How much did each book cost? Equation: (6.0 / 2.0) The above are samples of SVAMP data. Think step by step and give me a novel sample of SVAMP dataset.

Table 14: Sample prompts used for QNLI, RTE, WNLI, MRPC, and SVAMP dataset. We use the same prompt for different methods for fair comaprison.

Data	Method	Prompt
SQUAD	*GOLD*, ZeroGen, ProGen, P2Model	Below are samples of SQUAD data. It has a ’context’ which is a paragraph from wikipedia, a ’question’ from the paragraph and a short ’answer’ for the question. ’answers’ are directly from the ’context’. 1. context: Architecturally, the school has a Catholic character. Atop the Main Building’s gold dome is a golden statue of the Virgin Mary. Immediately in front of the Main Building and facing it, is a copper statue of Christ with arms upraised with the legend ’Venite Ad Me Omnes’. Next to the Main Building is the Basilica of the Sacred Heart. Immediately behind the basilica is the Grotto, a Marian place of prayer and reflection. It is a replica of the grotto at Lourdes, France where the Virgin Mary reputedly appeared to Saint Bernadette Soubirous in 1858. At the end of the main drive (and in a direct line that connects through 3 statues and the Gold Dome), is a simple, modern stone statue of Mary. question:To whom did the Virgin Mary allegedly appear in 1858 in Lourdes France? answer:Saint Bernadette Soubirous 2. context: As at most other universities, Notre Dame’s students run a number of news media outlets. The nine student-run outlets include three newspapers, both a radio and television station, and several magazines and journals. Begun as a one-page journal in September 1876, the Scholastic magazine is issued twice monthly and claims to be the oldest continuous collegiate publication in the United States. The other magazine, The Juggler, is released twice a year and focuses on student literature and artwork. The Dome yearbook is published annually. The newspapers have varying publication interests, with The Observer published daily and mainly reporting university and other news, and staffed by students from both Notre Dame and Saint Mary’s College. Unlike Scholastic and The Dome, The Observer is an independent publication and does not have a faculty advisor or any editorial oversight from the University. In 1987, when some students believed that The Observer began to show a conservative bias, a liberal newspaper, Common Sense was published. Likewise, in 2003, when other students believed that the paper showed a liberal bias, the conservative paper Irish Rover went into production. Neither paper is published as often as The Observer; however, all three are distributed to all students. Finally, in Spring 2008 an undergraduate journal for political science research, Beyond Politics, made its debut. question: In what year did the student paper Common Sense begin publication at Notre Dame? answer: 1987 3. context: The College of Engineering was established in 1920, however, early courses in civil and mechanical engineering were a part of the College of Science since the 1870s. Today the college, housed in the Fitzpatrick, Cushing, and Stinson-Remick Halls of Engineering, includes five departments of study – aerospace and mechanical engineering, chemical and biomolecular engineering, civil engineering and geological sciences, computer science and engineering, and electrical engineering – with eight B.S. degrees offered. Additionally, the college offers five-year dual degree programs with the Colleges of Arts and Letters and of Business awarding additional B.A. and Master of Business Administration (MBA) degrees, respectively. question: How many departments are within the Stinson-Remick Hall of Engineering? answer: fiven The above are samples of SQUAD data. Think step by step and give me a novel sample of SQUAD dataset.

Table 15: Sample prompts used for SQUAD and Adv-QA datasets. We use the same prompt for different methods for fair comaprison.

Method	ID	Random Samples
QNLI	1	**Sentence:** "The Titanic was designed to move at speeds of up to 21 knots and could accommodate more than 3,500 passengers and crew.", **Question:** "What was the maximum speed of the Titanic?", **Label:** "entailment"
	2	**Sentence:** "The new manager is very friendly and always willing to help.", **Question:** "How many years of experience does the new manager have in the field?", **Label:** "not_entailment"
RTE	1	**Sentence1:** "The use of digital technologies in agriculture has the potential to revolutionize food production and reduce the environmental impact of farming.", **Sentence2:** "The development of digital agriculture will lead to increased food availability and better nutrition for the world's population.", **Label:** "entailment"
	2	**Sentence1:** "There are several reasons behind the growing trend of veganism, including the health benefits of a plant-based diet, the ethical concerns surrounding animal welfare, and the environmental impact of livestock production.", **Sentence2:** "The consumption of dairy products leads to better bone health.", **Label:** "not_entailment"
MRPC	1	**Sentence1:** "The new AI technology has the potential to revolutionize healthcare.", **Sentence2:** "The latest advancement in artificial intelligence is expected to transform the medical field.", **Label:** "equivalent"
	2	**Sentence1:** "The company will launch a new product line next quarter.", **Sentence2:** "The company will lay off 10% of its workforce next month.", **Label:** "not_equivalent"
SVAMP	1	**Body**: "A car travels 480 kilometers in 8 hours.", **Question**: "How far does the car travel per hour?", **Equation**: " (480.0 / 8.0) ", **Answer**: "60"
	2	**Body**: "If Sally has 35 apples and she gives 10 to her friend.", **Question**: "How many apples does Sally have left?", **Equation**: " (35.0 - 10.0) ", **Answer**: "25"
NL4OPT	1	**Problem**: "A company produces two products, A and B, with different production rates. The company has a total of 10 machines, and each machine can produce either product. The production rate of product A is 2 machines per day, while the production rate of product B is 3 machines per day. The company wants to maximize the total daily production of both products." **Formulation**: Variables: x: number of machines producing product A y: number of machines producing product B Objective Function: maximize $2.00 * x + 3.00 * y$ Constraints: $x + y \leq 10.00$ $x \geq 0$ $y \geq 0$
	2	**Problem**: "A company produces two types of smartphones, A and B. Smartphone A has a battery life of 24 hours and a profit margin of \$50 per unit, while smartphone B has a battery life of 12 hours and a profit margin of \$30 per unit. The company has a total budget of \$100000 for producing these smartphones. The company wants to determine the number of units of each smartphone to produce to maximize the profit.", **Formulation**: Variables: x: number of units of smartphone A y: number of units of smartphone B Objective Function: maximize $(50.00 * x + 30.00 * y)$ Constraints: $x + y \leq 100000.00$

Table 16: Two data samples generated by *GOLD* randomly sampled from the generated dataset for different datasets.