ICLRegulator: Regulating In-Context Learning Behavior for Applicability Authorization

Anonymous ACL submission

Abstract

In-context learning (ICL) is one of the most popular capabilities of large language models (LLMs). This feature allows users to perform a new task without updating the model. Unlike conventional fine-tuning, ICL provides greater flexibility but also raises concerns. Users may apply models to arbitrary data, including improper or sensitive content, which can conflict with usage policies or the interests of model owners. To this end, we introduce the concept of "applicability authorization" tailored for LLMs, particularly for ICL behavior. We propose ICLRegulator, a fine-tuning framework with novel loss functions that control ICL selectively: the model produces random outputs on restricted data while maintaining correct outputs elsewhere. Empirical results show that our method disables ICL on targeted inputs without degrading ICL performance on unrestricted data or the model's overall capabilities.

1 Introduction

Natural language processing tasks, such as sentence classification and inference, are often solved by using a pre-trained model and fine-tuning it with the downstream dataset. Pre-trained language models like GPT (Radford et al., 2019) and BERT (Devlin et al., 2019) are trained on a huge amount of data from the internet through self-supervised methods. These models contain extensive knowledge across a wide range of topics, and fine-tuning them produces remarkable performance on the downstream task. Recently, LLMs have demonstrated powerful ICL capabilities (Brown et al., 2020; Min et al., 2022; Wei et al., 2023). This approach shifts the use of LLMs from traditional fine-tuning to direct application. Users can prompt the LLM with a set of input-label pairs and test input to perform the desired task, such as sentence classification and question answering (Min et al., 2022; Wei et al., 2023; Pan et al., 2023). ICL has the advantage over fine-tuning as it allows LLMs to "learn" new tasks during inference without the need for additional training. It saves human effort and resources while showcasing LLMs' ability to generalize from a few examples, similar to human capabilities.

However, ICL also presents opportunities and challenges. On one hand, it enables models to adapt to new tasks from just a few examples, offering substantial convenience. On the other hand, users may exploit ICL with content that violates model policies or conflicts with the model owner's interests. For example, minors could access age-restricted tasks, or users might process copyrighted or sensitive data through ICL. These scenarios not only raise an accountability risk for the model owner, potentially implicating them in unlawful activities, but also emphasize the importance of regulating ICL behavior. Importantly, this challenge differs from the issue of toxicity or bias in LMs (Sheng et al., 2019; Schick et al., 2021; Dinan et al., 2020; Gehman et al., 2020), which usually arises from biased training datasets and can be mitigated by improving data quality or adversarial training. By contrast, ICL neither depends on nor alters the model's training objectives or dataset, raising the question of how to regulate capabilities that were never explicitly trained. Drawing inspiration from (Wang et al., 2022), we address this concern by determining which data can be used for ICL, leading to the concept of "ICL applicability authorization."

Prior research on authorization in machine learning falls into two categories: model usage authorization (Chakraborty et al., 2020; Alam et al., 2020), which restricts access and deployment to authorized users, and applicability authorization (Wang et al., 2022), which determines what data may be used with a model. Our work aligns with the latter. (Wang et al., 2022) fine-tune a model optimized for the MNIST dataset with a dedicated patch and become ineffective for the USPS dataset, even though both datasets are used for digit

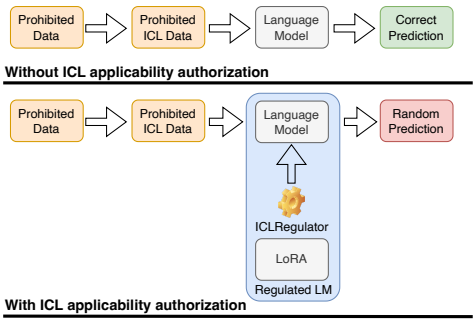


Figure 1: Overview of ICL applicability authorization.

recognition. Their research emphasizes permitting only specific data to be used on the model. In contrast, we intend to prohibit the use of ICL on target data while preserving the ability to use ICL on all other (non-target) data.

In this paper, we integrate applicability authorization within ICL to regulate ICL behavior for classification tasks on various datasets. We propose ICLRegulator, a flexible fine-tuning framework with LoRA (Hu et al., 2022) and optimized with three complementary loss functions. The disable loss suppresses ICL on target data, the maintenance loss preserves ICL performance on non-target data, and the utility loss ensures that outputs remain consistent with the original LLM when handling non-ICL prompts across both target and non-target data ICLRegulator fine-tunes the LoRA adapter with these loss functions to authorize the ICL applicability while preserving the integrity of the regulated LLM with respect to the original LLM, as shown in Figure 1.

Our empirical evaluations over 4 datasets (i.e., FP, SST-2, TREC, and AGnews) and 3 LLMs (i.e., LLaMA, OPT, and Cerebras) illustrate the efficacy of ICLRegulator in disabling the ICL ability on target data without compromising the ICL and regular LM function on all data. We further analyze the role of each loss function in controlling ICL behavior and evaluate strategies for generating data outside the data space of the target dataset to enhance performance. In summary, we make the following contributions:

- We propose ICLRegulator, the first fine-tuning framework to regulate ICL behavior on LLMs.
- We conduct a comprehensive evaluation showing that the regulated LLM selectively disables ICL on target data while maintaining functionality elsewhere.
- We explore diverse data generation strategies to improve the generalization of regulated LLMs.

2 Problem Statement

In our problem formulation, we have two parties, i.e., the model owner and users.

Model Owner. The model owner provides an LLM that is capable of performing ICL. For a standard prompt, the LLM produces a sequence of posterior probability vectors, which are then decoded into tokens using a chosen decoding strategy. With ICL, users can prompt the LLM by presenting a few demonstrations with the test input, and the LLM again generates posterior vectors from which labels can be inferred.

In classification tasks, users determine the label by selecting the token with the highest probability. However, this capability also enables misuse: ICL can be applied to data that violates model policies or conflicts with the model owner's interests. To mitigate this risk, the owner introduces ICL applicability authorization using ICLRegulator to regulate ICL behavior across datasets. We define three requirements for successful applicability authorization:

- **Requirement 1**: The regulated LLM must disable ICL on target data by producing uniform probability distributions over the label set.
- **Requirement 2**: The regulated LLM must preserve ICL on non-target data, yielding probability distributions similar to those of the original LLM.
- **Requirement 3**: The regulated LLM must retain the original LLM's functionality on standard (non-ICL) prompts.

Model Owner's Background Knowledge. To disable ICL on target data, we assume the owner has access to a small subset of that data, obtained either from usage logs or directly from the requester. Template and label choices can strongly influence ICL performance; however, we assume the owner has only limited knowledge of them. Accordingly, we adopt a minimal template and the default label set, as described in Appendix B.2.

Users. We consider two types of users: regular and malicious. All users have unrestricted access to the model and can issue either standard or ICL-formatted prompts. While regular users employ the model as intended, malicious users may attempt to apply ICL to prohibited data for personal gain, thereby violating usage policies or creating risks for the model owner.

3 Methodology

In this section, we present ICLRegulator, a fine-tuning framework for regulating ICL behavior in LLMs. We first describe its fundamental building block, followed by an overview of the complete fine-tuning procedure.

3.1 Building Blocks

Shadow Dataset. To disable ICL on target data, we require the regulated LLM to output random predictions when presented with target ICL prompts. This is equivalent to training the model to take a few input–label pairs with a test input and return a uniform probability vector. For this purpose, the owner first constructs a shadow ICL dataset $\mathcal{D}_{sdICL}$ that contains ICL prompts paired with labels.

Suppose the target dataset has size m. The owner samples k examples to form u unique demonstrations $(d_1, d_2, \ldots, d_u)$. Each target data is paired with all demonstrations $z_{ij} = (d_i, x_j)$, resulting in a shadow ICL dataset $\mathcal{D}_{sdICL} = \{z_{ij}, \cdots, z_{um}\}$, where $|\mathcal{D}_{sdICL}| = u \times m$.

Next, we assign labels to each z_{ij}. A naive approach is to use random or flipped one-hot labels, but using a one-hot label as the target label can lead to overshooting, which can destabilize training. Instead, we derive soft distorted labels $\hat{y}_{sdICL}$ by modifying the posterior distribution $P(z_{sdICL})$ from the original LLM θ.

Specifically, we set the probabilities of all known labels (e.g., "negative" and "positive") to zero, and renormalize to obtain $P(z_{sdICL})$. For example, in LLaMA-13B, the indices for "positive" (6374) and "negative" (8178) are zeroed out before constructing $\hat{y}_{sdICL}$ as presented:

$$\hat{y}_{sdICL} = \begin{cases} 0 & \text{if } i \in C \\ P^i(z_{sdICL}) & \text{if } i \notin C \end{cases}$$

where i is the token index from the vocabulary V. At this point, we have the shadow ICL dataset ready $\mathcal{D}_{sdICL} = \{(z_{sdICL}, \hat{y}_{sdICL})_i\}_{i=1}^{u \times m}$.

Surrogate Dataset. While the regulated LLM should output random predictions on target data, it must preserve correct behavior on non-target data presented in ICL format. Since non-target data refers to any input outside the target dataset, and the space of possible data is effectively unbounded, we approximate it using a surrogate dataset. Specifically, we sample m outputs from the original LLM and treat them as surrogate data.

To create the surrogate ICL input z_{sgICL}, we follow the same process of constructing the shadow ICL input by assembling u unique k input-label pairs with the input. Regarding the label y_{sgICL}, we can directly use the posterior obtained from the original LLM as the soft label and assemble the surrogate ICL dataset as $D_{sgICL} = \{(z_{sgICL}, y_{sgICL})_i\}_{i=1}^{u \times m}$.

However, random sampling may produce data semantically or structurally similar to the target dataset, causing overlap between $\mathcal{D}_{sdICL}$ and $\mathcal{D}_{sgICL}$ and potentially harming performance. To mitigate this, we explore alternative data generation strategies, as discussed in Section 5.3. In addition, the regulated LLM should function normally on regular sentences for all data. We build a surrogate dataset $\mathcal{D}_{sg}$, which is distinct from $\mathcal{D}_{sgICL}$, and defined as $\mathcal{D}_{sg} = \{x_1, x_2, \cdots, x_m\}$.

Disable Loss. The first component of ICLRegulator is the disable loss $\mathcal{L}_D$, which prevents the model from using target data for ICL. Given a data point $(z_{sdICL}, \hat{y}_{sdICL})$ from $\mathcal{D}_{sdICL}$, we can maximize the log-likelihood of $\hat{y}_{sdICL}$ to optimize ϕ while freezing θ as:

$$\mathcal{L}_D = \max_{\phi} log\ P(\hat{y}_{sdICL} | z_{sdICL}; \theta; \phi) \quad (1)$$

This objective drives the regulated LLM to assign uniform probabilities across all known labels when processing target data in ICL format.

Maintenance Loss. Prior studies found that fine-tuning models for a specific downstream task can adversely affect their performance on other tasks (Mukhoti et al., 2023). Our results in Section 5.2 also align with these findings. If we only fine-tune the model with $\mathcal{L}_D$, the ICL and regular LM function of the regulated LLM degenerate compared to the original LLM. Therefore, we present the maintenance loss, $\mathcal{L}_M$, which aims to retain the ICL ability on non-target data using $\mathcal{D}_{sgICL}$. Similar to how $\mathcal{L}_D$ operates, we also maximize the log-likelihood of y_{sgICL} to optimize ϕ while θ

remains fixed. The loss $\mathcal{L}_M$ is computed:

$$\mathcal{L}_M = \max_{\phi} log\ P(y_{sgICL}|z_{sgICL};\theta;\phi) \quad (2)$$

This objective ensures that the regulated LLM produces posteriors consistent with the original LLM on non-target ICL data, thereby maintaining its ICL performance.

Utility Loss. While $\mathcal{L}_D$ and $\mathcal{L}_M$ regulate ICL on target and non-target data, respectively, they do not address the model's ability to process regular (non-ICL) prompts. Similar to the ICL performance drops on non-target data, we also observe that the performance of processing regular sentences on the regulated LLM is compromised, likely due to "concept forgetting" (Mukhoti et al., 2023). Therefore, we introduce the utility loss $\mathcal{L}_U$ to re-align the regulated LLM with respect to the original LLM.

A straightforward approach is to fine-tune the model with $\mathcal{D}_{sg}$ using the autoregressive loss. However, the model might overfit the dataset since the size of the surrogate dataset is much smaller than the pre-training dataset. As an alternative, we minimize the distance between text representations from the regulated and original models inspired by (Mukhoti et al., 2023; Zhang et al., 2018). Specifically, for each token at position i in layer j, we extract hidden states h_i^j and h'^j_i from the original and regulated models. To strengthen alignment, we extend this across all l layers, normalize each vector with the L_2-norm, and concatenate them. The objective is to minimize the L2 distance between these sets of representations, and thus, the loss $\mathcal{L}_A$ is defined as:

$$\mathcal{L}_U = \min_{\phi} ||h' - h||_2 \quad (3)$$

This encourages the regulated LLM to produce similar outputs as the original LLM on all data.

3.2 Overall

After presenting the building blocks, we introduce the overall process of ICLRegulator. The owner aims to fine-tune the original LLM to regulate the ICL behavior. For each epoch, ICLRegulator samples the data from $\mathcal{D}_{sdICL}$ and obtains the output from the regulated LLM to compute the cross-entropy for $\mathcal{L}_D$. It computes the cross-entropy for $\mathcal{L}_M$ with the data from $\mathcal{D}_{sgICL}$. Also, ICLRegulator samples the data from $\mathcal{D}_{sg}$ to obtain a sequence of normized vector $\{h'_1, h'_2, \cdots, h'_n\}$ from the regulated LLM and $\{h_1, h_2, \cdots, h_n\}$ from the original LLM, where n is the sentence length. Then, ICLRegulator concatenates them and computes $\mathcal{L}_U$ by minimizing the L2 distance between h' and h. The final objective function is:

$$\mathcal{L} = \mathcal{L}_D + \mathcal{L}_M + \mathcal{L}_U \quad (4)$$

This optimization is repeated for T epochs, progressively aligning the regulated LLM with the desired ICL applicability behavior.

4 Experimental Setup

4.1 Datasets and Models

We conduct experiments on five datasets. FP (Malo et al., 2014) contains financial news sentences labeled as positive, negative, or neutral. SST-2 (Wang et al., 2019) consists of movie reviews with binary sentiment labels, making it a natural comparison point with FP when disabling ICL on one but not the other. TREC (Li and Roth, 2002) is a question classification benchmark with six categories: abbreviation, entity, description, human, location, and number. AGNews (Zhang et al., 2015) is a topic classification dataset with four classes: world, sports, business, and science & technology. Lambada (Paperno et al., 2016) evaluates LMs' ability in processing narrative texts. In our setup, FP, SST-2, TREC, and AGNews each serve as the target dataset in turn, while the others act as auxiliary sources. We evaluate on 100 examples per dataset.

For models, we use LLaMA (Touvron et al., 2023), OPT (Zhang et al., 2022), and Cerebras (Dey et al., 2023). OPT and Cerebras are English-focused, whereas LLaMA is multilingual. Among 13B-parameter models, LLaMA demonstrates stronger reasoning, comprehension, and generation abilities (Fourrier et al., 2024), with competitive zero- and few-shot performance. We therefore focus on LLaMA-13B, while also reporting results for its 7B and 30B variants.

4.2 Implementation Details

Training. To regulate ICL behavior, we primarily experiment with LoRA on top of the LLaMA-13B through the paper unless specifically stated. For LoRA, the rank is set at 8, alpha at 32, and the dropout rate is 0.1. We use a batch size of 4 and a learning rate of 1 x 10^{-4} with the Adam optimizer. We train the model for 20 epochs. [1]

[1] We also explored training longer, but it did not improve performance.

Target Dataset		Auxiliary Datasets			
Target	Accuracy (%)	FP (%)	SST-2 (%)	TREC (%)	AGnews (%)
FP	0.31 (-0.49)	-	0.80 (-0.13)	0.68 (-0.02)	0.82 (-0.08)
SST-2	0.41 (-0.52)	0.71 (-0.09)	-	0.68 (-0.02)	0.88 (-0.02)
TREC	0.13 (-0.57)	0.76 (-0.04)	0.93 (+0.00)	-	0.89 (-0.01)
AGnews	0.21 (-0.69)	0.62 (-0.18)	0.92 (-0.01)	0.71 (-0.01)	-

Table 1: The ICL performance and changes of the regulated LLM.

Target Dataset		Auxiliary Datasets				
Target	Δ PPL	FP	SST-2	TREC	AGnews	Lambada
FP	+0.020	-	-2.164	+0.024	-0.004	-0.012
SST-2	+2.885	+0.091	-	+0.172	+0.006	0.012
TREC	+0.044	-0.004	-2.190	-	-0.007	+0.002
AGnews	+0.003	+0.025	-2.006	+0.131	-	+0.006

Table 2: The utility changes of the regulated LLM.

Test Set. Despite the success of ICL, its performance is unstable, depending on the choice of template and the combination of demonstrations. While many prior studies have explored the factors and potential mitigation approaches, the study is beyond the scope of this paper. For simplicity, we utilize the minimal template as shown in Appendix B.2. To build test sets, we randomly sample demonstrations from the training data using random seeds 40–50. For each dataset, we retain the demonstration yielding the best performance and exclude those sampled examples from training.

Other Details. By default, all experiments use $k = 16$ input–label pairs and $u = 40$ unique demonstrations. For target data, 100 examples are sufficient to deactivate ICL. For the utility loss, we extract text representations from every $l = 2$ transformer layers.

4.3 Evaluation Metrics

ICL Performance. We evaluate ICL performance on both target and auxiliary datasets. For the target dataset, we compute test accuracy; performance close to random guessing indicates successful disabling of ICL. For auxiliary datasets (i.e., datasets distinct from the target), we also compute test accuracy. Accuracy comparable to the original LLM indicates preserved ICL ability on non-target data.

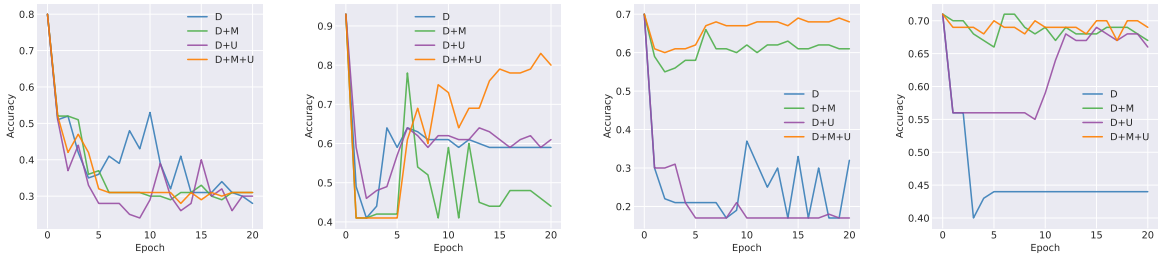
Utility. To assess general LM utility, we measure perplexity (PPL). Since our focus is on non-ICL prompts, the regulated LLM should exhibit PPL values close to those of the original model. We therefore report the difference $\Delta PPL = PPL_{original} - PPL_{regulated}$. A small ΔPPL indicates that the regulated LLM retains the utility of the original.

5 Evaluation

We analyze ICLRegulator with three different studies. First, we evaluate the overall performance of the regulated LLM when disabling ICL on different target datasets. Next, we examine the contribution of each loss function and its impact. Finally, we investigate alternative strategies for constructing the surrogate dataset. For reference, Table 13 reports the baseline ICL performance and utility of LLaMA-13B on each dataset.

5.1 General

We first evaluate the ICL performance of the regulated LLM after disabling the ICL ability on target data (i.e., FP, SST-2, TREC, and AGnews). Table 1 illustrates the accuracy and changes compared to the original LLM. In general, the regulated LLM can disable the ICL ability on target data and bring the accuracy close to random guessing. Meanwhile, the ICL performance on auxiliary datasets only drops negligibly overall. For instance, when the target dataset is TREC, its accuracy drops from 70% to 13%, while the accuracy on FP, SST-2, and AGnews is maintained at 76%, 93%, and 89%, respectively. However, we observe that the ICL performance on the auxiliary dataset is affected in



(a) FP (b) SST-2 (c) TREC (d) AGnews

Figure 2: The ICL performance changes during fine-tuning under different loss combinations. D = disable loss, M = maintenance loss, U = utility loss.

some cases. When the target object is FP, the accuracy on SST-2 drops by 13%, which is slightly higher than other auxiliary datasets. This unexpected result could be due to the overlapping label set between FP and SST-2, and deactivating the ICL ability on FP might also affect SST-2 accidentally. We observe the same phenomenon when targeting SST-2 as the accuracy on FP drops by 9%.

Next, we evaluate the utility of the regulated LLM. As shown in Table 2, the PPL changes are negligible across target and auxiliary datasets. The result demonstrates that the regulated LLM can produce outputs almost identical to the original on regular prompts. Note that the PPL change on SST-2 is generally higher since it has a higher PPL on the original LLM. Overall, these results indicate the efficacy of ICLRegulator to fine-tune the regulated LLM on deactivating the ICL ability on target data without compromising the ICL performance on non-target data and the regular LM function on all data simultaneously. In the following, we primarily use FP as the target dataset to investigate the effectiveness of ICLRegulator from other aspects.

5.2 Impact of the Loss Function

In this section, we investigate the impact of each loss function on controlling the ICL ability and regular LM ability on target and auxiliary data. In Table 3 and Table 4, we first note that using the disable loss alone can deactivate the ICL ability on FP completely from 80% to 28%. As expected, the ICL ability on auxiliary datasets is affected significantly. For instance, SST-2, TREC, and AGnews have a 34%, 38%, and 67% accuracy drop, respectively. The PPL change on them is also massive, with more than 20. All these results reflect that the regulated LLM has altered completely.

Then, when we optimize the model with both disable and maintenance loss, the ICL ability on auxiliary datasets recovers to a certain extent without sacrificing the deactivating performance. For instance, TREC and AGnews have only 9% and 29% accuracy drops now, and the PPL changes are also much lower. Although the overall performance is not the best, the regulated LLM can function adequately close to the original LLM on data in non-ICL format. Besides, we note that both the ICL performance and utility on SST-2 are affected significantly. As mentioned in Section 5.1, the result is likely to be related to the overlapping label set. Therefore, fine-tuning the model with both disable and maintenance loss together might hurt the optimization.

On the other hand, we find that the utility changes across datasets are negligible when using both disable and utility loss. It confirms that the utility loss effectively preserves the regulated LLM's functionality in processing regular data. In-

	Target	Auxiliary		
Loss	FP (%)	SST-2 (%)	TREC (%)	AGnews (%)
D	-0.52	-0.34	-0.38	-0.67
D+M	-0.49	-0.49	-0.09	-0.29
D+U	-0.50	-0.32	-0.53	-0.70
D+M+U	-0.49	-0.13	-0.02	-0.08

Table 3: The ICL performance changes with different loss setups.

	Target	Auxiliary			
Loss	FP	SST-2	TREC	AGnews	Lambada
D	+24.08	-198.5	+20.75	+48.89	+32.12
D+M	-0.484	-18.46	-0.192	-0.062	-0.099
D+U	+0.004	-2.220	-0.039	-0.007	-0.006
D+M+U	+0.020	-2.164	+0.024	-0.004	-0.012

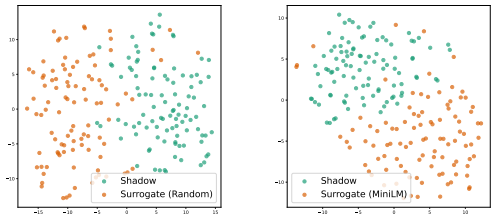
Table 4: The utility changes with different loss setups.

	Target	Auxiliary		
Method	FP (%)	SST-2 (%)	TREC (%)	AGnews (%)
Random	-0.49	-0.13	-0.02	-0.08
Auxiliary (SST-2)	-0.62	+0.00	-0.02	-0.06
JS	-0.50	-0.12	+0.00	+0.00
M-Cos	-0.66	-0.10	-0.01	+0.00
M-L2	-0.65	-0.09	-0.01	+0.00
L-Cos	-0.49	-0.50	+0.00	+0.00
L-L2	-0.63	-0.24	-0.06	-0.04

Table 5: The ICL performance changes with different generation strategies. M = MiniLM, L = LLaMA.

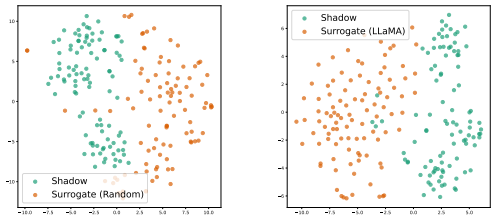
	Target	Auxiliary			
Method	FP	SST-2	TREC	AGnews	Lambada
Random	+0.020	-2.164	+0.024	-0.004	-0.012
Auxiliary (SST-2)	+0.000	+0.332	+0.105	+0.000	+0.019
JS	+0.022	+3.085	+0.005	-0.009	-0.001
M-Cos	-0.005	+2.814	+0.178	+0.002	-0.001
M-L2	+0.011	+0.468	+0.244	+0.001	-0.008
L-Cos	+0.022	+3.187	+0.009	+0.002	-0.001
L-L2	+0.004	+3.058	+0.028	+0.002	+0.012

Table 6: The utility changes with different strategies.



(a) Random (b) MiniLM-L2

Figure 3: The t-SNE visualization on the shadow and surrogate data with MiniLM using L2 distance.



(a) Random (b) LLaMA-L2

Figure 4: The t-SNE visualization on the shadow and surrogate data with LLaMA-13B using L2 distance.

terestingly, we observe that the ICL performance on SST-2 is better than using the maintenance loss as the accuracy goes up to 61%. Taken together, using all three losses produces the best performance in controlling ICL behavior and maintaining the LLM functionality.

Fine-Tuning Process. Furthermore, we report the ICL performance changes over the fine-tuning process in Figure 2. First, with no surprise, all loss combinations can disable the ICL ability on FP in the early stage, as shown in Figure 2a. Second, we observe that the ICL ability on the auxiliary dataset is affected at the beginning of the fine-tuning, but recovers later. For instance, the accuracy on TREC drops initially and is recovered after 5 epochs, as depicted in Figure 2c. Besides, we observe that the performance on SST-2 fluctuates when using only disable and maintenance loss during the fine-tuning in Figure 2b. It confirms our hypothesis that disable and maintenance loss conflict with each other, leading to unstable optimization.

5.3 Surrogate Dataset Augmentation

In Section 3, we build the surrogate dataset by random sampling data from the original LLM. However, such samples may resemble target data, which can hinder optimization. To justify our hypothesis, we first assume that we have access to the SST-2 dataset and use it to replace the surrogate dataset. In Table 5 and Table 6, we observe that the ICL performance of the regulated LLM is now identical to the original LLM, as the accuracy change on SST-2 is 0%. The results demonstrate that a good surrogate dataset can reduce the impact of fine-tuning. Since auxiliary data are unavailable in practice, we instead generate surrogate samples designed to differ from both target and previously sampled data by maximizing their distance. We explore distance at two levels: discrete tokens and latent representations.

Discrete Token. At the token level, we use Jaccard Similarity (JS) to measure the distance between target and sample data in terms of tokens. JS computes the overlapping ratio between two text sentences: $JS = \frac{x_1 \bigcap x_2}{x_1 \bigcup x_2}$. It evaluates how closely the two sentences are contextually related by calculating the proportion of common tokens to total tokens. The idea here is to generate data with tokens different from the target and other sample data. In detail, we retain only the data with a JS score of less than 0.1, meaning a maximum of 10% token overlap between the target and sample data. In Table 5, we find that it can improve the ICL ability on TREC and AGnews, resulting in a 0% accuracy drop. In addition, the utility remains decent as the PPL changes are still negligible across datasets.

Latent Representation. At the representation

level, we extract sentence embeddings using LLaMA (conditional LM) and MiniLM [2] (Masked LM). For MiniLM, we obtain the sentence representation by applying the mean pooling on top of the token representation. For LLaMA, we use the representation of the last token, which attends to the full sequence. We then measure the sentence distance with the Euclidean distance (L2) or cosine similarity (Cos) and only keep the generated data when the score is larger or smaller than the threshold, respectively.[3]

Results in Table 5 show that MiniLM-based generation consistently outperforms LLaMA. For example, with MiniLM, using L2 or Cos reduces FP accuracy to 14% and 15% while mitigating ICL drops on TREC and AGNews; using L2 further preserves SST-2 ICL performance at 84%. By contrast, LLaMA-based generation offers limited improvements and can degrade auxiliary datasets (e.g., Cos improves TREC/AGNews but lowers SST-2 to 69%). In terms of utility, MiniLM with L2 achieves the best results, yielding a ΔPPL of 0.468 on SST-2—comparable to using FP as the surrogate dataset (Table 6). Finally, visualizations in Figure 3 and Figure 4 confirm that MiniLM-based L2 sampling covers broader latent space with less overlap between surrogate and shadow data.

6 Related Works

6.1 In-Context Learning

ICL was first observed by (Brown et al., 2020), showing that LLMs can perform new tasks without parameter updates. (von Oswald et al., 2023) suggested that ICL approximates gradient-based few-shot learning during the forward pass, while (Xie et al., 2022) interpreted ICL as implicit Bayesian inference arising from interactions between the pre-training distribution and hidden Markov models. (Min et al., 2022) further examined factors such as the role of ground-truth labels in ICL performance. Overall, prior work has focused on understanding and enhancing ICL. In contrast, we investigate for controlling ICL behavior.

6.2 Model Authorization

Model authorization was originally proposed to protect intellectual property. (Alam et al., 2020) encrypted network parameters with a secret key, while (Chakraborty et al., 2020) derived keys from device fingerprints to restrict model use to specific hardware. More recently, (Wang et al., 2022) introduced applicability authorization for data-centric protection, which specifies what data a model may process rather than who may access it. For example, restricting an MNIST model from performing reliably on USPS. We extend this concept to ICL, aiming to regulate ICL behavior by determining what data an LLM can process through ICL.

7 Discussions

Resistance to Input Modification. In practical scenarios, users have considerable flexibility in how they construct ICL prompts, including choices of templates, label sets, and the number of demonstrations. Such freedom raises the question of whether regulated LLMs remain robust when these factors are adapted. We evaluate the resistance of ICLRegulator to such modifications by systematically varying templates, labels, and demonstration counts. Appendix A.1 shows that the regulated LLM maintains its intended behavior under these conditions, indicating that ICLRegulator is resilient to input-level manipulation.

Hyperparameters and Extensions. We also investigate the effect of different hyperparameters and system setups. Specifically, we test ICLRegulator across models of varying sizes, deactivate ICL on multiple target datasets simultaneously, vary the number of demonstrations and sampled data, and compare against existing fine-tuning methods. Appendix A shows that ICLRegulator scales effectively and regulates ICL without degrading overall performance. Also, we have extended our study to generative tasks and have demonstrated effectiveness in regulating ICL on translation and instruction tasks, as shown in Appendix A.3.

8 Conclusion

This paper introduces ICLRegulator, the first fine-tuning framework for regulating ICL behavior in LLMs. Our method selectively disables ICL on target data while preserving both ICL on non-target data and the model's general language capabilities. Experiments demonstrate that ICLRegulator effectively mitigates risks of misuse, offering a practical mechanism for safeguarding LLM deployment. We believe these findings provide useful guidance for model owners and open new directions for research on controllable ICL.

[2] https://www.sbert.net/index.html

[3] We determine the threshold by measuring the score between data points in the target dataset.

Limitations

Generative Tasks. This work focuses on regulating ICL behavior in classification tasks. While the proposed framework demonstrates effectiveness in this setting, extending ICLRegulator to generative tasks (e.g., summarization, translation, or open-ended text generation) presents additional challenges. In such cases, the notion of "target" versus "non-target" data is less clearly defined, and enforcing applicability authorization may require new formulations of loss functions or evaluation metrics. In the future, we will explore more complex generation tasks.

Robustness Studies. Our robustness analysis primarily examines variations in template, label set, and number of demonstrations. However, real-world deployments may face more sophisticated prompting strategies, such as paraphrased prompts or cross-lingual demonstrations. A more complicated study is needed to evaluate ICLRegulator under these complex conditions and to further strengthen its guarantees against different prompting.

Ethical considerations

In this study, we use public datasets and do not involve human participants. Therefore, our Institutional Review Boards (IRB) do not classify this work as human subjects research. In this work, our goal is to control the ICL behavior, and it does not involve using any harmful content.

References

https://github.com/tatsu-lab/stanford_alpaca.

Manaar Alam, Sayandeep Saha, Debdeep Mukhopadhyay, and Sandip Kundu. 2020. Deep-Lock: Secure Authorization for Deep Neural Networks. *CoRR abs/2008.05966*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language Models are Few-Shot Learners. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. NeurIPS.

Abhishek Chakraborty, Ankit Mondal, and Ankur Srivastava. 2020. Hardware-assisted intellectual property protection of deep learning models. *IACR Cryptol. ePrint Arch.*, page 1016.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep Reinforcement Learning from Human Preferences. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 4299–4307. NIPS.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186. ACL.

Nolan Dey, Gurpreet Gosal, Zhiming Chen, Hemant Khachane, William Marshall, Ribhu Pathria, Marvin Tom, and Joel Hestness. 2023. Cerebras-GPT: Open Compute-Optimal Language Models Trained on the Cerebras Wafer-Scale Cluster. *CoRR abs/2304.03208*.

Emily Dinan, Angela Fan, Adina Williams, Jack Urbanek, Douwe Kiela, and Jason Weston. 2020. Queens are Powerful too: Mitigating Gender Bias in Dialogue Generation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8173–8188. ACL.

Wikimedia Foundation. Acl 2019 fourth conference on machine translation (wmt19), shared task: Machine translation of news.

Clémentine Fourrier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. 2024. Open llm leaderboard v2. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models. *CoRR abs/2009.11462*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations (ICLR)*.

Xin Li and Dan Roth. 2002. Learning Question Classifiers. In *International Conference on Computational Linguistics (COLING)*. ACL.

Pekka Malo, Ankur Sinha, Pekka J. Korhonen, Jyrki Wallenius, and Pyry Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *J. Assoc. Inf. Sci. Technol.*, pages 782–796.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the Role of Demonstrations: What Makes In-Context Learning Work? In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 11048–11064. ACL.

Jishnu Mukhoti, Yarin Gal, Philip H. S. Torr, and Puneet K. Dokania. 2023. Fine-tuning can cripple your foundation model; preserving features may be the solution. *CoRR abs/2308.13320*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. NeurIPS.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. 2023. What In-Context Learning "Learns" In-Context: Disentangling Task Recognition and Task Learning. *CoRR abs/2305.09731*.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, page 1525–1534. ACL.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI blog*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *International Conference on Learning Representations (ICLR)*. ICLR.

Timo Schick, Sahana Udupa, and Hinrich Schütze. 2021. Self-Diagnosis and Self-Debiasing: A Proposal for Reducing Corpus-Based Bias in NLP. *Transactions of the Association for Computational Linguistics*.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The Woman Worked as a Babysitter: On Biases in Language Generation. In *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3405–3410. ACL.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. *CoRR abs/2302.13971*.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. 2023. Transformers Learn In-Context by Gradient Descent. In *International Conference on Machine Learning (ICML)*, pages 35151–35174. PMLR.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *International Conference on Learning Representations (ICLR)*.

Lixu Wang, Shichao Xu, Ruiqi Xu, Xiao Wang, and Qi Zhu. 2022. Non-Transferable Learning: A New Approach for Model Ownership Verification and Applicability Authorization. In *International Conference on Learning Representations (ICLR)*.

Jerry W. Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. 2023. Larger language models do in-context learning differently. *CoRR abs/2303.03846*.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An Explanation of In-context Learning as Implicit Bayesian Inference. In *International Conference on Learning Representations (ICLR)*. ICLR.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–595. IEEE.

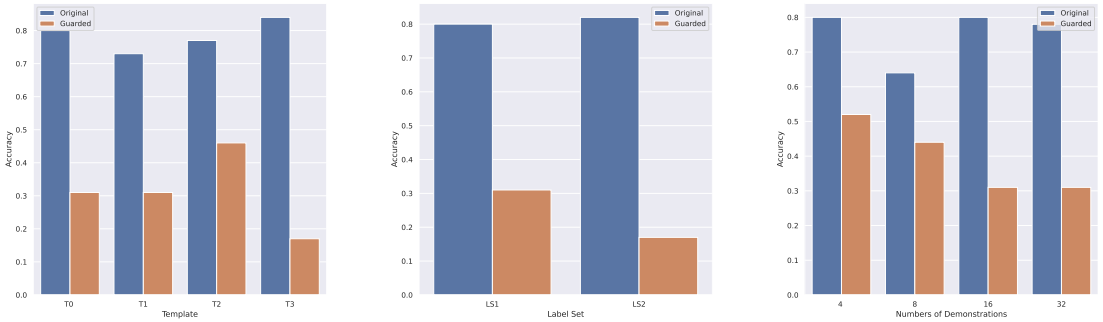
Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona T. Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: Open Pre-trained Transformer Language Models. *CoRR abs/2205.01068*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 649–657. NIPS.

A Additional Study

A.1 Robustness

In section 3 and section 4, we present the details of constructing the shadow and surrogate ICL dataset and set up the demonstration (16 examples) using the minimal template with the default label set. In practical scenarios, the user has the freedom to decide the use of the template, label set, and number of demonstrations. If the malicious user realizes that some demonstration setups are ineffective, they can switch to another setup. In this section, we evaluate the robustness of the regulated LLM against



(a) Different templates. (b) Different label sets. (c) Different numbers of demonstrations.

Figure 5: The ICL performance on FP against the adaptive adversary.

these adaptive demonstrations by considering three variables: template, label, and number of demonstrations.

Template. First, we conduct our experiment by using the other three templates in Appendix B.2. As depicted in Figure 5a, the regulated LLM remains effective at deactivating the ICL ability on FP, even when encountering templates that are not presented during the fine-tuning phase. We observe that the ICL ability on FP with template 1 and 3 still has over 40% and 60% accuracy drops, respectively. On the other hand, using template 2 can perverse the ICL performance slightly up to 46%, but that is not enough to be usable for the adversary.

Label. Second, we investigate the implications of label mismatches using other label sets in Appendix B.2. Apart from altering the template, modifying the label is another viable avenue for adversaries. In Figure 5b, we find that switching labels does not undermine the ICL applicability authorization. Instead, the accuracy on FP drops further from 31% to 17%.

Number of Demonstrations. While our study constructs the shadow and surrogate ICL dataset utilizing 16 demonstrations, an adversary might employ a different quantity (e.g., 4, 8, or 32). We conduct the experiment to investigate the impact of varying the number of demonstrations. As shown in Figure 5c, reducing the number of demonstrations marginally weakens the regulated LLM's ability to deactivate ICL on FP. For instance, the accuracy increases from 31% to 52% as the number of demonstrations decreases from 16 to 4. But still, the adversary should not be able to use ICL to perform the task with low accuracy.

	Target	Auxiliary		
Models	FP (%)	SST-2 (%)	TREC (%)	AGnews (%)
LLaMA-13B	0.31 (-0.49)	0.80 (-0.13)	0.68 (-0.02)	0.82 (-0.08)
OPT-13B	0.17 (-0.56)	0.86 (-0.07)	0.59 (+0.03)	0.30 (-0.48)
Cerebras-13B	0.31 (-0.56)	0.81 (-0.08)	0.40 (-0.01)	0.37 (-0.40)

Table 7: The ICL performance with different models.

	Target	Auxiliary		
Models	FP (%)	SST-2 (%)	TREC (%)	AGnews (%)
7B	0.51 (-0.26)	0.92 (-0.03)	0.77 (-0.03)	0.85 (-0.04)
13B	0.31 (-0.49)	0.80 (-0.13)	0.68 (-0.02)	0.82 (-0.08)
30B	0.31 (-0.52)	0.91 (-0.04)	0.92 (+0.00)	0.91 (+0.00)

Table 8: The ICL performance with different model sizes of LLaMA model.

A.2 Hyperparameter

Models. We start by evaluating the efficacy of ICLRegulator across different models: LLaMA-13B, OPT-13B, and Cerebras-13B. The objective is to ascertain that ICLRegulator can be applied universally across varied LLMs. We provide the original ICL performance and utility for OPT and Cerebras in subsection B.3. In Table 7, all regulated LLMs successfully reduce the ICL performance on FP to nearly random guessing. We observe that both OPT and Cerebras have better ICL performance (less performance drops) on SST-2 with 86% and 81% accuracy, respectively. Besides, they exhibit a higher accuracy drop in using ICL on AGnews, achieving only 30% and 37% accuracy. This result may be explained by the fact that AGnews data is similar to the pre-training data of these models. In Table 18 (subsection B.4), AGnews generally has lower perplexity, and even mirror changes to the model with AGnews data would likely affect the model. But still, the overall performance is

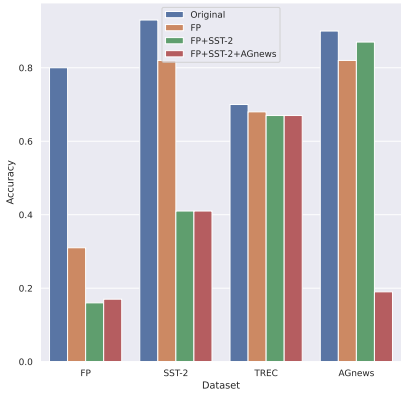


Figure 6: The ICL performance of deactivating multiple target datasets. The target dataset is presented in the legend.

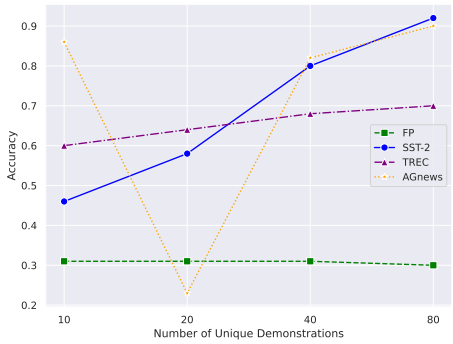


Figure 7: The ICL performance with different numbers of unique demonstrations.

comparable to the original model.

Model Size. We then extended our evaluation by varying the model size of LLaMA from 7B to 30B, as shown in Table 8. We provide the original ICL performance and utility for the 7B and 30B models in subsection B.3. We observed that the smaller model has a weaker capability to deactivate the ICL ability on FP. In contrast, the 13B and 30B models can completely deactivate the ICL ability on FP. Since the fine-tuning effect of ICLRegulator on the 7B model is smaller, it also has a smaller impact on SST-2 and AGnews. On the other hand, the 30B model is able to deactivate the ICL ability on FP without compromising the ICL performance on other auxiliary datasets, including SST-2. Table 19 (subsection B.4) reports the utility changes, demonstrating that the impact of ICLRegulator on the 7B and 30B models is less significant.

Multiple Target. In reality, the model owner might need to deactivate multiple data simultaneously. Therefore, we study the effectiveness of ICLRegulator in deactivating multiple datasets in two scenarios: 1) FP and SST-2, and 2) FP, SST-2, and AGnews. As shown in Figure 6 and Table 20, ICLRegulator successfully deactivates multiple targets without compromising the model utility. In the first scenario, the accuracy of FP and SST-2 drops to 16% and 41%, respectively, while the accuracy of TREC and AGnews only decreases by 3% and 3%. In the second scenario, the accuracy of FP, SST-2, and AGnews drops to 17%, 41%, and 19%, respectively. Although there is a slight change in the accuracy and utility of TREC in both scenarios, the general ICL performance and model utility remain comparable to the original model. The results indicate the effectiveness of using ICLRegulator to manage multiple ICL control requirements.

Number of Unique Demonstrations. We evaluate the effect of varying the number of unique demonstrations from 10 to 80. Typically, increasing the number of unique demonstrations should enhance the model's ability to control ICL behavior on both target and non-target data. In Figure 7, the results reveal that even with a smaller number of unique demonstrations, it is still possible to deactivate the ICL ability on FP. However, we also suspect that performance drops at 20 demonstrations are due to the instability of ICL, and using a larger number of unique demonstrations can mitigate the issue. In addition, using more unique demonstrations can preserve the ICL ability on other datasets better. For instance, the accuracy on SST-2 increases from 46% to 90%, and for TREC, it increases from 60% to 70%. These relationships may be explained by using more unique demonstrations, which enable the regulated LLM to better distinguish the data from SST-2 or FP during fine-tuning. In Table 21 (subsection B.4), we observe that the PPL change on TREC decreases as the number of unique demonstrations increases, although the difference is negligible.

Sampling Size. In addition to selecting the number of unique demonstrations, the owner can also choose to sample more data from the original LLM. We investigate the impact of varying sampling sizes (100, 200, 400) and believe this study can further enhance the results in subsection 5.3. In Figure 8, we observe that using more sample data can enhance the deactivating effect on FP, with the ICL performance dropping to 17% when using 200 and

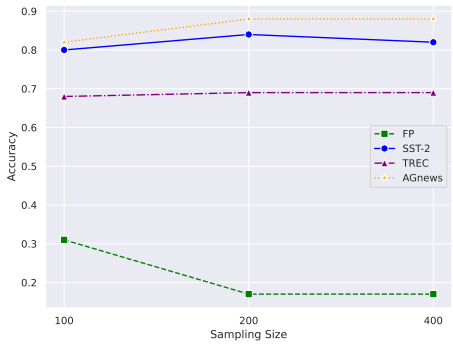


Figure 8: The ICL performance with different sampling sizes.

	Target	Auxiliary		
# Sample	FP	SST-2	TREC	AGnews
ICLRegulator	0.31 (-0.49)	0.80 (-0.13)	0.68 (-0.02)	0.82 (-0.08)
SFT	0.21 (-0.59)	0.82 (-0.11)	0.59 (-0.11)	0.26 (-0.64)
DPO	0.03 (-0.77)	0.41 (-0.52)	0.43 (-0.27)	0.22 (-0.68)

Table 9: The ICL performance changes with different fine-tuning methods.

400 sample data. Also, using more sample data can slightly improve the ICL performance on TREC and AGnews. In Table 22 (subsection B.4), we also observe that the PPL change on SST-2 is lower with more sampling. For instance, the PPL change is 0.339 and 0.399 when the sampling size is 200 and 400, respectively. Generally, utilizing more sampled data can enhance the ICL controllability of the regulated LLM, but it also demands additional computational resources.

Fine-tuning Methods. To control the ICL behavior, one straightforward approach is to fine-tune the model to produce an incorrect prediction given the ICL prompt. Another approach is to fine-tune LLM with reinforcement learning from human feedback, which has become a popular method for aligning models with human values (Christiano et al., 2017). Therefore, we compare ICLRegulator with Supervised Fine-Tuning (SFT) (Ouyang et al., 2022) and Direct Preference Optimization (DPO) (Rafailov et al., 2023) approaches on deactivating the ICL for the FP dataset (training details in subsection B.5). As shown in Table 9 and Table 23, we report the ICL performance and the model utility.

First, we observe a sharp decline in FP accuracy after applying SFT, showing the success of deactivating the target data. However, there is a varying degree of decrease in accuracy across other non-target datasets, along with notable changes in

	Auxiliary					
Target	WMT19	Alpaca	FP (%)	SST-2 (%)	TREC (%)	AGnews (%)
Baseline	18.24	19.00	0.80	0.93	0.70	0.90
WMT19	1.262	18.96	0.82	0.94	0.70	0.89
Alpaca	17.83	7.75	0.81	0.92	0.69	0.89

Table 10: The ICL performance of the regulated LLM on deactivating different generation tasks.

the model utility. We suspect this is due to significant domain shifts or distortions in the model, which could potentially lead to "concept forgetting." In fact, we can consider that training with disable loss is a softer version of SFT, where the model is trained to maximize the log-likelihood of the softly distorted label. In contrast, SFT aims to maximize the log-likelihood of the incorrect one-hot label. Therefore, training with SFT will likely distort the model significantly, similar to the case where only disable loss is applied, as discussed in subsection 5.2. Besides, the result shows that optimizing the model with the one-hot incorrect label will heavily alter it, which demonstrates the need of the soft distorted label.

Second, we find that DPO can deactivate the ICL ability for FP without affecting the model utility. However, the ICL performance for other data is affected. For instance, the accuracy drops to 41% for SST-2, 43% for TREC, and 22% for AGnews. This result follows a similar trend observed in subsection 5.2 where only disable and utility losses are applied to train the model. It suggests the need to use the maintenance loss to preserve the ICL ability on non-target data.

A.3 Extension to Generative Task

In the previous section, we have shown that our ICLRegulator is effective in controlling the ICL behavior for classification tasks. In practice, the model owner might want to prevent the model from being used for specific generative tasks via ICL, such as generating propaganda. Although different alignment methods have been proposed to fine-tune the model with human value, we have shown that such methods do not work well for ICL data. In this section, we investigate the possibility of extending ICLRegulator for generative tasks.

Setup. For instance, some users might translate improper information into another language or attempt to get an answer that is undesired or harmful to the public. To conduct the study for the generative task, we use the WMT19 (Foundation) and the Alpaca dataset (sta). We use examples of the

model owner wanting to prevent users from using the model for English-to-German translation or instruction answering via ICL. Specifically, we evaluate ICLRegulator on the LLaMA-13B model using the same ICL setup as section 4 with the template "{sentence} \n Output:". To build the shadow ICL dataset in section 3, we set the output logit of the first tokens of each ground truth output to 0. This ensures that the regulated model cannot generate the correct translation or complete the instruction. We compute the BLEU score using the NLTK package [4] to evaluate the translation performance and the ROUGE score to evaluate the answer given instructions.

Results. First, we compute the BLEU score for performing WMT19 on LLaMA-13B without ICL. Specifically, we use the template "Translate the sentence to German: sentence Output:" in a zero-shot manner and obtain a BLEU score of 7.17. Then, we perform the translation with 16 demonstrations and achieve a BLEU score of 18.24, as shown in Table 10. This substantial increase indicates the efficiency and potential of ICL to enhance the model's performance in handling unseen generative tasks. Next, we evaluate the performance of ICLRegulator in deactivating ICL for the WMT19 data. We observe that the BLEU score drops significantly to 1.262. This decrease illustrates the effectiveness of ICLRegulator in preventing the model from performing the English-to-German translation task. Meanwhile, the model's utility and ICL performance for other tasks remains comparable to the original performance, as shown in Table 24.

We next report the ROUGE scores for the Alpaca dataset. Initially, we also evaluate its zero-shot performance and obtain a score of 16.00. Subsequently, the ROUGE score improves to 19.00 with 16 demonstrations. But still, we are able to deactivate ICL with ICLRegulator for the Alpaca data, and it leads to a decline in the score to 7.75. The drop in performance is less pronounced compared to WMT19, likely due to the diverse nature of tasks included in the Alpaca dataset, such as writing and translation. This diversity also explains the reason for the minor drops in WMT19. Nevertheless, the model's utility and ICL performance for other tasks remain largely consistent with its original capabilities. These results demonstrate that ICLRegulator effectively restricts specific unwanted generative tasks while preserving the model's overall performance across a range of other tasks.

B Additional Experiment Details

B.1 Experimental Setups

All models and datasets in our experiments are downloaded from HuggingFace. LLaMA (non-commercial license), OPT (MIT), and Cerebras (Apache License 2.0) are each distributed under their respective original licenses. The SST2, FP, TREC, AGNews, Lambda, WMT16, and Alpaca datasets are released under Creative Commons licenses. All experiments in the paper were conducted on an A100 (80GB) compute node.

B.2 Prompt Templates

For each task, we use the minimum template and default label set. For FP, we use three extra templates for the robustness study. All prompts and labels are presented in Table 11 and Table 12.

B.3 Baseline Performance

For the model and size study, we provide the original ICL performance and utility for OPT-13B, Cerebras-13B, LLaMA-7B, and LLaMA-30B in Table 14, Table 15, Table 16, and Table 17, respectively.

B.4 More Results

We report the utility performance for Appendix A, including different models, different LLaMA model sizes, different numbers of unique demonstrations, and different sampling sizes in Table 18, Table 19, Table 21, and Table 22.

B.5 Training Details

For SFT and DPO, we use the same setup as section 4 for fine-tuning the LoRA. We set the rank to 8, alpha to 32, and the dropout rate to 0.1. In addition, we use a batch size of 4 and a learning rate of 1 x 10^{-4} with the Adam optimizer. For SFT, we randomly select an incorrect label as the continuous on the target ICL prompt, and truncate the input length to 400. For DPO, we set the ground label as the rejected answer, and randomly select an incorrect label as the chosen answer. In addition, we quantize the model to 4 bits. We implement the training using Huggingface's TRL package [5].

[4] https://www.nltk.org/

[5] https://github.com/huggingface/trl

Template #	Example
0	<Sentence> Label: <Label>
1	<Sentence> Sentiment: <Label>
2	<Sentence> The sentiment is <Label>
3	<Sentence> The sentiment of the text is <Label>

Table 11: List of templates taken from (Pan et al., 2023).

Dataset	Label #	
FP	1 2	negative/neutral/positive terrible/neutral/good
SST-2	1	terrible/good/neutral
TREC	1	expression/entity/description/ human/location/number
AGnews	1	World/Sports/Business/Science

Table 12: List of default labels for each dataset.

Dataset	Accuracy (%)	PPL
FP	0.80	31.42
SST-2	0.93	405.0
TREC	0.70	51.92
AGnews	0.90	11.25
Lambada	-	27.64

Table 13: The original ICL performance and utility of LLaMA-13B for each dataset.

Dataset	Accuracy (%)	PPL
FP	0.73	98.01
SST-2	0.93	443.3
TREC	0.56	129.1
AGnews	0.78	25.81
Lambada	-	37.41

Table 14: The original ICL performance and utility of OPT-13B for each dataset.

Dataset	Accuracy (%)	PPL
FP	0.55	100.3
SST-2	0.89	350.1
TREC	0.41	345.1
AGnews	0.77	33.81
Lambada	-	44.46

Table 15: The original ICL performance and utility of Cerebras-13B for each dataset.

Dataset	Accuracy (%)	PPL
FP	0.77	32.28
SST-2	0.95	572.9
TREC	0.80	52.61
AGnews	0.89	12.05
Lambada	-	28.39

Table 16: The original ICL performance and utility of LLaMA-7B for each dataset.

Dataset	Accuracy (%)	PPL
FP	0.83	28.75
SST-2	0.95	509.9
TREC	0.92	51.18
AGnews	0.91	10.66
Lambada	-	26.45

Table 17: The original ICL performance and utility of LLaMA-30B for each dataset.

Model	FP	SST-2	TREC	AGnews	Lambada
LLaMA-13B	31.44 (+0.020)	402.8 (-2.164)	51.95 (+0.024)	11.24 (-0.004)	27.63 (-0.012)
OPT-13B	87.12 (-10.89)	417.9 (-25.35)	132.3 (+3.124)	25.62 (-0.186)	39.14 (+1.728)
Cerebras-13B	100.2 (-0.077)	349.8 (-0.293)	344.7 (-0.417)	33.80 (-0.014)	44.46 (-0.003)

Table 18: The utility and changes with different models.

Model	FP	SST-2	TREC	AGnews	Lambada
7B	32.27 (-0.011)	571.0 (-1.924)	52.86 (+0.252)	12.05 (+0.006)	28.40 (+0.004)
13B	31.44 (+0.020)	402.8 (-2.164)	51.95 (+0.024)	11.24 (-0.004)	27.63 (-0.012)
30B	28.74 (-0.006)	511.4 (+1.538)	51.19 (+0.008)	10.66 (+0.007)	26.46 (+0.014)

Table 19: The utility and changes with different sizes of LLaMA model.

Target	FP	SST-2	TREC	AGnews	Lambada
FP	+0.020	-2.164	+0.024	-0.004	-0.012
FP+SST-2	-0.006	-2.189	+0.101	-0.005	+0.049
FP+SST-2+AGnews	+0.014	+0.437	+0.212	+0.006	+0.041

Table 20: The utility changes with different target settings on the LLaMA model.

	Target		Auxiliary		
# Unique	FP	SST-2	TREC	AGnews	Lambada
10	+0.016	-2.168	+0.104	+0.005	+0.013
20	+0.010	-1.955	+0.074	-0.004	+0.008
40	+0.020	-2.164	+0.024	-0.004	-0.012
80	+0.022	+0.509	+0.068	-0.006	+0.034

Table 21: The utility changes with different numbers of unique demonstrations.

	Target		Auxiliary		
# Sample	FP	SST-2	TREC	AGnews	Lambada
100	+0.020	-2.164	+0.024	-0.004	-0.012
200	+0.004	+0.339	+0.006	+0.063	-0.003
400	-0.017	+0.399	-0.051	+0.001	+0.002

Table 22: The utility changes with different sampling sizes.

	Target		Auxiliary		
# Sample	FP	SST-2	TREC	AGnews	Lambada
ICLRegulator	+0.020	-2.164	+0.024	-0.004	-0.012
SFT	+608.5	+881.1	inf	+313.2	inf
DPO	+0.415	+19.20	+0.504	+0.086	-0.185

Table 23: The utility changes with different fine-tuning methods.

	FP	SST-2	TREC	AGnews	Lambada
Original	31.42	405.0	51.92	11.25	27.64
Regulated	31.44	405.4	52.14	11.25	27.64

Table 24: The utility of the regulated LLM on deactivating English-to-German translation.