STAR: Constraint LoRA with Dynamic Active Learning for Data-Efficient Fine-Tuning of Large Language Models

Linhai Zhang[♠♢*] **Jialong Wu**[♠♢*] **Deyu Zhou**[♠♢†] **Guoqiang Xu**[♡]

[♠]School of Computer Science and Engineering, Southeast University, Nanjing, China
[♢]Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, China
[♡]SANY Group Co., Ltd.

{lzhang472, jialongwu, d.zhou}@seu.edu.cn
xuguoqiang-2012@hotmail.com

Abstract

Though Large Language Models (LLMs) have demonstrated the powerful capabilities of few-shot learning through prompting methods, supervised training is still necessary for complex reasoning tasks. Because of their extensive parameters and memory consumption, both Parameter-Efficient Fine-Tuning (PEFT) methods and Memory-Efficient Fine-Tuning methods have been proposed for LLMs. Nevertheless, the issue of large annotated data consumption, the aim of Data-Efficient Fine-Tuning, remains unexplored. One obvious way is to combine the PEFT method with active learning. However, the experimental results show that such a combination is not trivial and yields inferior results. Through probe experiments, such observation might be explained by two main reasons: uncertainty gap and poor model calibration. Therefore, in this paper, we propose a novel approach to effectively integrate uncertainty-based active learning and Low-Rank Adaptation (LoRA). Specifically, for the uncertainty gap, we introduce a dynamic uncertainty measurement that combines the uncertainty of the base model and the uncertainty of the full model during the iteration of active learning. For poor model calibration, we incorporate the regularization method during LoRA training to keep the model from being over-confident, and the Monte-Carlo dropout mechanism is employed to enhance the uncertainty estimation. Experimental results show that the proposed approach outperforms existing baseline models on three complex reasoning tasks.[1]

1 Introduction

Large Language Models (LLMs) (Brown et al., 2020; Wei et al., 2021; OpenAI, 2022; Touvron et al., 2023a,b; Zhao et al., 2023) have demonstrated the powerful capabilities of zero/few-shot learning with prompting techniques, including In-Context Learning (Dong et al., 2022) and Chain-of-Thought (Wei et al., 2022), where no parameter update is required. However, previous studies (Hendrycks et al., 2020; Yuan et al., 2023; Bai et al., 2023; Isik et al., 2024) have shown that further fine-tuning is still crucial for tasks involving complex reasoning such as arithmetic reasoning (Roy and Roth, 2016; Cobbe et al., 2021) and commonsense reasoning (Mihaylov et al., 2018; Clark et al., 2019), *etc.*

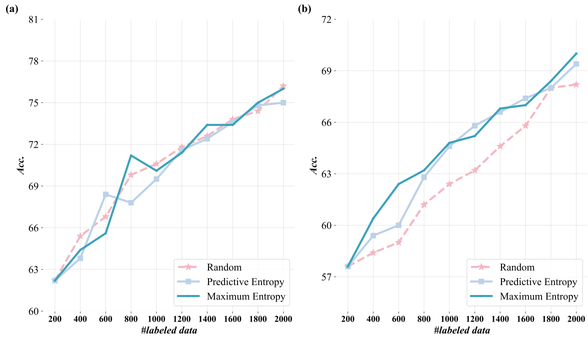

Figure 1: (a) Active learning combined with LoRA compared to passive learning. (b) Active learning combined with full parameter tuning compared to passive learning.

Fine-tuning LLMs requires updating a large number of parameters, which takes a lot of time and consumes considerable memory. Taking LLaMA-7B (Touvron et al., 2023a) as an example, fine-tuning it on a dataset of 52k instances takes over 12 hours on 4 A100 80G GPUs (Bommasani et al., 2021; Taori et al., 2023). Therefore, Parameter Efficient Fine-Tuning (PEFT) methods (Houlsby et al., 2019; Lester et al., 2021; Li and Liang, 2021; Hu et al., 2021; Ding et al., 2023) and Memory Efficient Fine-Tuning (MEFT) methods (Liao et al., 2023) have been proposed. In addition to updating a vast number of parameters and consuming substantial memory, a neglected factor in LLMs fine-tuning is the extensive consumption of annotation data. Moreover, due to the inherent complexity of tasks, the human annotation resources required

[*]Equal Contribution.
[†]Corresponding Author.
[1]Our code and results will be available at https://github.com/callanwu/STAR.

Findings of the Association for Computational Linguistics: ACL 2024, pages 3519–3532
August 11-16, 2024 ©2024 Association for Computational Linguistics

for fine-tuning LLMs are also significant (Ouyang et al., 2022).

Therefore, it is important to develop the Data-Efficient Fine-Tuning (DEFT) method for LLMs. A common practice to improve data efficiency is active learning (Cohn et al., 1996; Settles, 2009), while it has been shown that the PEFT methods can alleviate the reliance on annotated data to some extent (Ding et al., 2023). A straightforward idea for DEFT is to combine the PEFT method with active learning. However, such a combination is not trivial. As shown in Figure 1 (a), simply fine-tuning an LLM with LoRA (Hu et al., 2021) under an uncertainty-based active learning framework yields consistently inferior performances compared to passive learning (random selection of data in active learning) on the OpenBookQA dataset, while as shown in Figure 1 (b), fine-tuning LLM with full parameter under active learning yields performances better than passive learning.

To investigate the mechanism behind this uncommon phenomenon, probe experiments are conducted by investigating the prediction confidence and entropy of the LLM with LoRA during uncertainty-based active learning. Based on the experimental results, we deduce two potential reasons for this phenomenon. The first issue is ***uncertainty gap***. To be more specific, the uncertainty calculated for selecting data during active learning comes from the full parameters, while only partial parameters remain tuned during PEFT. It suggests that the conventional way of calculating uncertainty may not reflect the knowledge required by the PEFT parameters and therefore undermines the performance of active learning. The second issue is ***poor model calibration*** which becomes particularly significant when using the PEFT method (Wang et al., 2023). It further indicates that the uncertainty calculated in the conventional way is not well-calibrated, and the data selected for active learning becomes sub-optimal.

To address the aforementioned issues, we propose con**S**train**T** LoRA with dynamic **A**ctive lea**R**ning (**STAR**), a novel approach to effectively integrate uncertainty-based active learning and LoRA. Specifically, for the uncertainty gap, we introduce a dynamic uncertainty measurement that combines the uncertainty of the base model and the uncertainty of the full model during the iteration of active learning. For poor model calibration, we incorporate the regularization method during LoRA training to keep the model from being over-confident, and the Monte-Carlo dropout mechanism (Gal and Ghahramani, 2016) is employed to enhance the uncertainty estimation. Experimental results show that the proposed approach outperforms existing baseline models on three complex reasoning tasks. The above issues are partially resolved.

In conclusion, our contributions are three-fold:

- As far as we know, we are the first to investigate and uncover the reasons why directly combining active learning with LoRA fails to achieve comparable performance with passive learning through probe experiments.
- A novel DEFT method, **STAR**, is proposed to effectively combine PEFT with active learning through criterion revision and model regularization.
- Extensive experimental results show that the proposed method addresses the issues and outperforms other baselines.

2 Related Work

2.1 Efficient Fine-tuning Methods

As LLMs continue to expand in size, the computational and financial resources required for fine-tuning these models become increasingly prohibitive. To address this challenge, Efficient Fine-Tuning has emerged as an essential area of research (Wan et al., 2023). The methods can be classified into PEFT and MEFT (Wan et al., 2023; Liao et al., 2023). PEFT, in particular, aims to adjust a minimal subset of the model's parameters, thus conserving computational resources while maintaining or enhancing model performance (Hu et al., 2023). We classify PEFT methods into four categories: Prompt Tuning (Lester et al., 2021; Liu et al., 2023b) only allow an additional k tunable tokens per downstream task to be prepended to the input text. Prefix Tuning (Li and Liang, 2021; Liu et al., 2022) keeps language model parameters frozen but optimizes a small continuous task-specific vector that pretends to be key-value pairs. Adapter (Houlsby et al., 2019; He et al., 2021) is a new module added between layers of a pre-trained network, which is a bottleneck architecture. Low-Rank Adaptation (LoRA) (Aghajanyan et al., 2021; Hu et al., 2021) reduces the parameters and enhances computational efficiency by applying low-rank matrices. Moreover, minimizing memory usage in fine-tuning for improving efficiency has also emerged as a critical topic (Liao et al., 2023), with several innovative solutions being proposed. Among these, techniques such as QLoRA (Dettmers et al., 2023), QA-LoRA (Xu et al., 2023), and LoftQ (Li et al., 2023) stand out for their ability to significantly reduce memory requirements without compromising model performance. In this paper, we focus on the application of LoRA.

2.2 Active Learning with LLMs

Active Learning (AL) has been extensively investigated across a multitude of NLP tasks, encompassing machine translation (Miura et al., 2016; Zhao et al., 2020), natural language inference (Snijders et al., 2023), named entity recognition (Shen et al., 2017) and text classification (Ein-Dor et al., 2020; Margatina et al., 2022; Schröder et al., 2023). In the era of LLMs, active learning is primarily employed in the selection of prompts and the annotation of data (Zhang et al., 2023b; Liu et al., 2023a; Xiao et al., 2023). For instance, Margatina et al. (2023) explores various active learning strategies for selecting the most relevant examples for in-context learning with LLMs. Diao et al. (2023) introduces an active prompting method that leverages uncertainty metrics to select questions for annotation. In the domain of integrating PEFT with AL, Jukić and Snajder (2023a) explored PEFT methods with different active learning in Pre-trained language models (PLMs) such as BERT (Devlin et al., 2019), demonstrating that the integration of PEFT with active learning can offer substantial performance gains. Different from Jukić and Snajder (2023a), we apply decoder-only generative LLMs as the backbones, to our knowledge, we are the first to integrate LLMs combined PEFT with AL within the realm of reasoning tasks.

3 Preliminaries

3.1 Parameter Efficient Fine-tuning

Parameter-Efficient Fine-Tuning (PEFT) methods aim to fine-tune only a small set of external parameters while keeping the backbone model frozen and achieving comparable or even superior performance (Hu et al., 2023). The main-steam PEFT methods include the adapter-based methods (Houlsby et al., 2019; He et al., 2021), prefix tuning (Li and Liang, 2021), and LoRA (Hu et al., 2021), among which LoRA is the most effective and widely used. In this paper, we mainly implement PEFT with LoRA.

LoRA (**Lo**w-**R**ank **A**daptation) assumes that the updation of the model weight matrix during training is low-ranked, which can be decomposed as the multiplication of two low-rank matrices.

$$\Delta W = \alpha BA \tag{1}$$

where ΔW is the updation of the model weight matrix, $B \in \mathcal{R}^{d \times r}$ and $A \in \mathcal{R}^{r \times k}$ are matrices of rank r, and α is constant scaling factor.

During training, the model weight matrix $\boldsymbol{W}$ is fixed and only ΔW is optimized. It is worth noticing that commonly A is randomly initialized and B is zero-initialized. In this way, we have $\boldsymbol{W} = \boldsymbol{W} + \Delta W$ at the beginning of training and the fine-tuned model is identical to the base model.

3.2 Active Learning

Active Learning (AL) methods aim to select informative examples from the data pool to maximize the performance with the required data budget or minimize the data budget to achieve the required performance. The family of AL methods mainly includes uncertainty-based methods (Lewis, 1995; Gal and Ghahramani, 2016), diversity-based methods (Sener and Savarese, 2018), and discriminative-based methods (Gissin and Shalev-Shwartz, 2019), where uncertainty-based methods are widely used and easy for implementation.

In our study, we mainly consider three AL strategies, including RANDOM selection as a passive learning baseline and two uncertainty-based criteria. Maximum Entropy (Lewis, 1995) and Predictive Entropy (Duan et al., 2023; Kadavath et al., 2022) are both based on uncertainty, but the former is label independent while the latter is label dependent. The key idea behind uncertainty-based AL methods is that models will learn more efficiently from examples in which they are difficult to predict and have high prediction uncertainty.

LLMs (Touvron et al., 2023b) inherently generate sentences in a free-form and auto-regressive fashion. This process entails the sequential prediction of the probability distribution for the subsequent token in a sentence. Let x represent the input prompt, and s denote the sentence generated by the LLM, comprising N tokens in total. For any given LLM, the probability of producing a specific token z_i as the i-th element in the sentence can be mathematically expressed as $p(z_i|s_{<i}, x)$, where

$1 \leq i \leq N$. Here, $s_{<i}$ symbolizes the sequence of previously generated tokens $\{z_1, z_2, ..., z_{i-1}\}$.
MAXIMUM ENTROPY(ME) is characterized by its independence from golden response. It quantitatively evaluates the *uncertainty* in a model's predictions by computing the entropy across all possible outcomes, formulated as:

$$ME(s, x) = \\ -\sum_{i=1}^{N}\sum_{j=1}^{V} p(v_{ij}|s_{<i}, x) \log p(v_{ij}|s_{<i}, x) \quad (2)$$

where s is the generated response, $p(v_{ij}|s_{<i}, x)$ is the probability of j-th token in vocabulary at i-th element in s, V is the vocabulary size.
PREDICTIVE ENTROPY(PE) incorporates golden response dependency, offering a measure of the expected information gain from the true label, given the predictive distribution. It is formulated as:

$$\begin{aligned} PE(\overline{s}, x) &= -\log p(\overline{s}|x) \\ &= \sum_{i=1}^{N} -\log p(\overline{z}_i|\overline{s}_{<i}, x) \end{aligned} \quad (3)$$

where $\overline{s}$ is the golden response, $p(\overline{z}_i|\overline{s}_{<i}, x)$ is the probability of i-th token in the golden response.

4 Probing PEFT on Prediction Uncertainty

In this section, we describe how to design probe experiments to investigate the reason behind the failure of LoRA combined with AL methods. We will first introduce the experiment setup, and then we will talk about how to prob LoRA under the AL framework with prediction confidence and prediction entropy. We also discuss how to conclude from the experimental results.

4.1 Probe Experiment Design

As uncertainty-based AL methods mainly depend on the confidence or uncertainty of model predictions to select examples during each iteration, it is straightforward to probe the confidence and uncertainty of model predictions during AL iterations.

We mainly focus on two training variants of LLMs. The first one, denoted as *PEFT* method, is LLM finetuned with LoRA, which is shown to be problematic under the AL framework. The second one, denoted as *Few-shot* method, is untuned LLM, which is taken as a control group. To enhance the performance of untuned LLM on downstream tasks, we employ In-Context Learning (Dong et al., 2022) prompting by adding demonstrations with the input prompt. LLaMA-2 (Touvron et al., 2023b) serves as the backbone LLM. Experiments are conducted on the BoolQ dataset (Clark et al., 2019) because its labels only include "*true*" and "*false*", which makes the prediction uncertainty and prediction confidence easy to calculate.

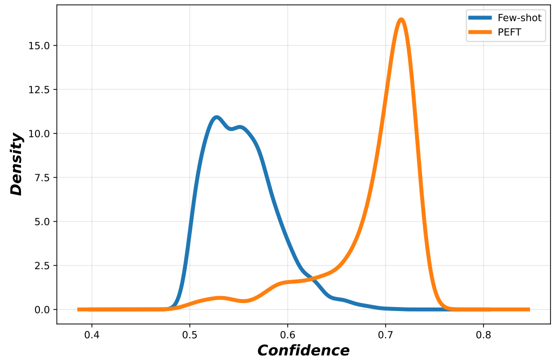

Figure 2: Density plot of confidence for wrong predictions.

4.2 Probing with Prediction Confidence

The first probe experiment is designed to explore whether the model prediction confidence of the *PEFT* method exhibits issues compared to *Few-shot* methods. The prediction confidence CF is measured by the maximum between the output probability on token "*true*" and "*false*".

$$CF = max(p_{true}, p_{false}) \quad (4)$$

where p_{true} and p_{false} denotes the probabilities of token "*true*" and "*false*", respectively.

Then the prediction confidence of *PEFT* and *Few-shot* are calculated and the density plot is drawn to make a comparison between these two methods as shown in Figure 2. To mitigate differences in model accuracy, we only consider the confidence of the model for the **wrong** predictions on the test set of BoolQ. The intuition is that for examples that the model is less likely to predict right, they should be less confident.

The *PEFT* method achieves an accuracy of 73.36%, which is much higher than the *Few-shot* method with an accuracy of 45.41%. As shown in Figure 2, the *PEFT* method is overconfident compared to the *Few-shot* method, where the confidence of the wrong prediction is as high as 70%, which indicates a ***model calibration issue***.

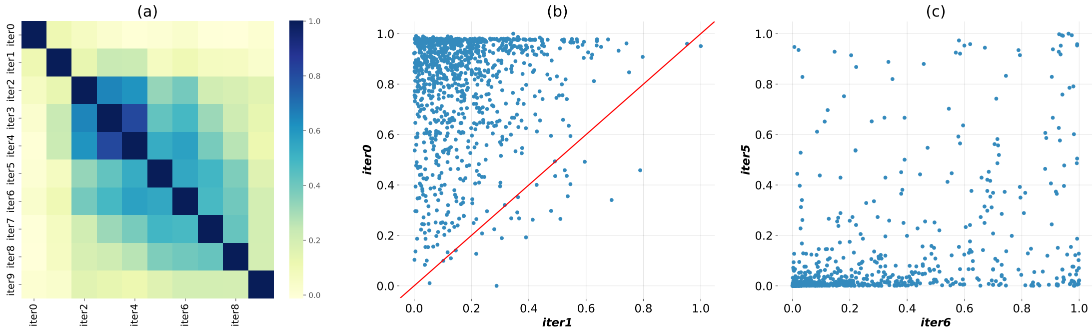

Figure 3: (a) Heatmap of correlation between prediction entropy across different iterations; (b) Scatter plot for prediction entropy between base model (Iter0) and model after first iteration (Iter1); (c) Same as (b), except values are taken from Iter5 and Iter6.

4.3 Probing with Prediction Entropy

The second probe experiment is designed to investigate the change of prediction entropy of *PEFT* model during active learning iteration. The MAXIMUM ENTROPY(ME) is employed as the uncertainty during active learning. Nine rounds of iteration are performed with 500 examples selected during each iteration. The *PEFT* model is trained with 500 examples at the beginning as a warm-up.

The correlation between examples with top 1000 entropy at the beginning is calculated and the heatmap of the correlation is shown in Figure 3 (a). As we can observe in Figure 3 (a), the correlation between the base model (model without PEFT tuning) and models after AL iteration is close to 0, which indicates ***a clear gap between the base model and PEFT model***. This phenomenon is even clear with the scatter plot in Figure 3, where the dots in Figure 3 (b) should appear around the red line but appear in the upper triangular region. In Figure 3 (c), the correlation coefficients of entropy between the two iterations become relatively normal, which is consistent with Figure 3 (a), suggesting that the gap between iterations has been alleviated.

5 Methods

In this section, we introduce the proposed method **STAR** in detail. We will first describe the overall workflow of **STAR**, then we will discuss methods to address the ***uncertainty gap*** issue and the ***model calibration*** issue. Finally, we will conclude the proposed method with a pseudocode.

5.1 LoRA under Active Learning Iteration

As shown in Figure 4, the k-th iteration of **STAR** consists of the following steps.

1. **Model Inference** that employs the present model M_k to make inference on unlabeled dataset D_k^U.
2. **Data Querying** that selects the most informative examples to form a subset S_k^U with the results of inference based on the dynamic uncertainty estimation method.
3. **Data Labeling** that labels the unlabeled subset S_k^U to form the labeled subset S_k^L.
4. **Dataset Updating** that updates the labeled dataset D_k^L by appending appending the labeled subset $D_{k+1}^L = D_k^L \cup S_k^L$.
5. **Model Training** that updates the present model with new labeled dataset D_{k+1}^L to get model M_{k+1} for next iteration.

5.2 Dynamic Uncertainty Measurement

To address the issue of ***uncertainty gap***, we proposed a dynamic uncertainty measurement to integrate the uncertainty of the frozen LLM (base model) and the uncertainty of LLM fine-tuned with LoRA (full model) dynamically based on the AL iteration.

The key idea is that at the beginning of PEFT training, the extra parameters are under-fitting, where the uncertainty calculated is less reliable than the frozen parameters. As the iteration of active learning increases, the uncertainty of the full model becomes more reliable, which is similar to the zero-initialized attention weight in LLaMA-adapater (Zhang et al., 2023a).

$$\mu = \lambda(t)\mu_b + (1 - \lambda(t))\mu_f \qquad (5)$$

where μ_b and μ_f denote the prediction uncertainty of the base model and the full model respectively,

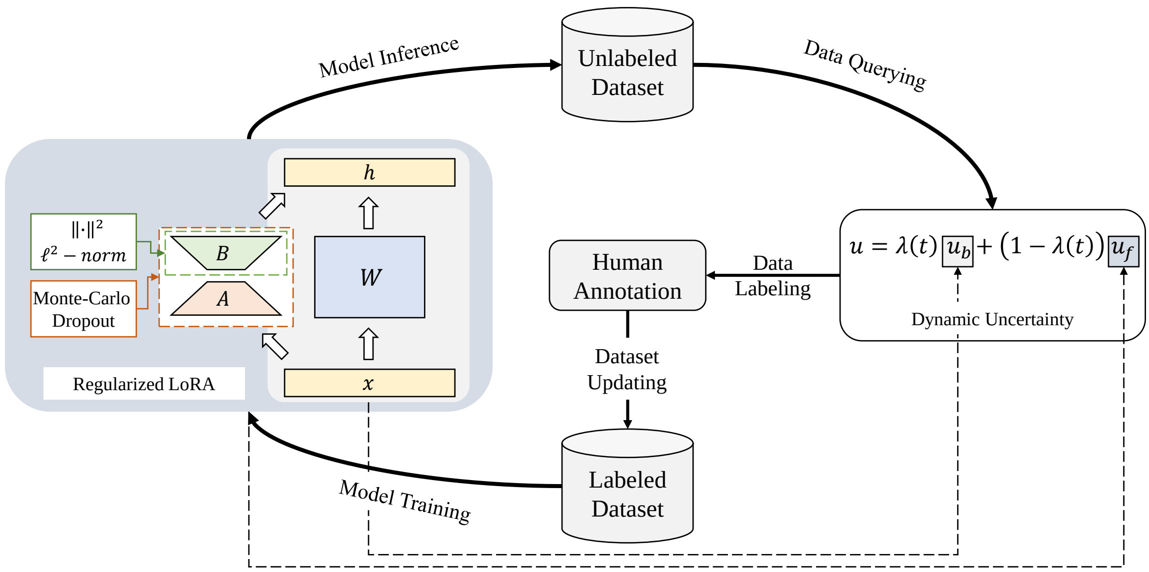

Figure 4: The framework of **STAR**. It primarily consists of five steps: Model Inference, Data Querying, Data Labeling, Dataset Updating, and Model Training.

$\lambda(t) \in [0, 1]$ is a monotone decreasing function of AL iteration t. Note that, our measurement approach only requires one additional computation of the base model at the beginning, which remains constant throughout and does not significantly increase the FLOPs.

5.3 Calibration with Hybrid Regularization

To address the issue of ***poor model calibration***, we propose a hybrid regularization method during PEFT training. As discussed in Section 4.2, the PEFT model demonstrates a pronounced tendency toward over-confidence, which indicates that the model is over-fitting.

Common approaches to prevent the model from being over-fitting include early-stoping (Doan and Liong, 2004), regularizations (Santos and Papa, 2022), and ensemble methods (Ganaie et al., 2022). Considering the difference between LoRA parameters A and B, we integrate two regularization methods into a hybrid regularization to keep LoRA from being over-fitting.

For the B matrix, which is zero-initialized, a L^2 norm weight decay is employed.

$$B_t \leftarrow B_{t-1} - \gamma(g_{t-1} - \beta B_{t-1}) \tag{6}$$

where g_{t-1} denotes the normalized gradient acquired from the standard Adam optimizer, and β denotes the strength of regularization.

For the A matrix, which is randomly Gaussian initialized $N(0, 1)$, the Monte-Carlo dropout (MC dropout) (Gal and Ghahramani, 2016) is adopted for more robust uncertainty estimation. MC dropout works by activating the dropout unit both in the training and inference stages, which can be regarded as an approximation to the Bayesian Neural Network. With the dropout unit activated during the inference stage, neural networks can generate different outputs with the same input, where expectations can be taken for more robust estimation.

$$\begin{aligned} \mu_f &= \frac{1}{K}\sum_k \mu_f^{(k)} \\ \mu_f^{(k)} &= \mathit{ME}(\mathrm{LLM}(x|\hat{A}_k, \hat{B}_k)) \end{aligned} \tag{7}$$

where K denotes the number of feedforward propagations during the inference stage, $\mu_f^{(k)}$ denotes the uncertainty estimated at k-th feedforward, $\hat{A}_k$ and $\hat{B}_k$ denote the LoRA matrices sampled from A and B with dropout unit activated.

5.4 Overall Algorithm

The overall algorithm of **STAR** is shown in Algorithm 1.

6 Experiments

6.1 Datasets

In our research, we employ three benchmark datasets spanning two categories of reasoning problems for AL evaluation:

Arithmetic Reasoning: the GSM8K

Method	GSM8K		BoolQ		OpenBookQA	
	AUC	RIPL	AUC	RIPL	AUC	RIPL
RANDOM	27.37	-	60.46	-	63.44	-
PREDICTIVE ENTROPY	27.30	-0.09	58.39	-5.24	63.05	-1.07
w/ **STAR**	<u>28.40</u>	<u>1.42</u>	<u>61.84</u>	<u>3.49</u>	<u>64.86</u>	<u>3.88</u>
MAXIMUM ENTROPY	27.16	-0.28	60.65	0.48	63.36	-0.22
w/ **STAR**	**28.83**	**2.01**	**61.91**	**3.67**	**66.17**	**7.47**

Table 1: The performance of different methods in a passive learning setup in terms of the AUC and RIPL. The optimal results among all methods are **bolded** and the second-best results are <u>underlined</u>.

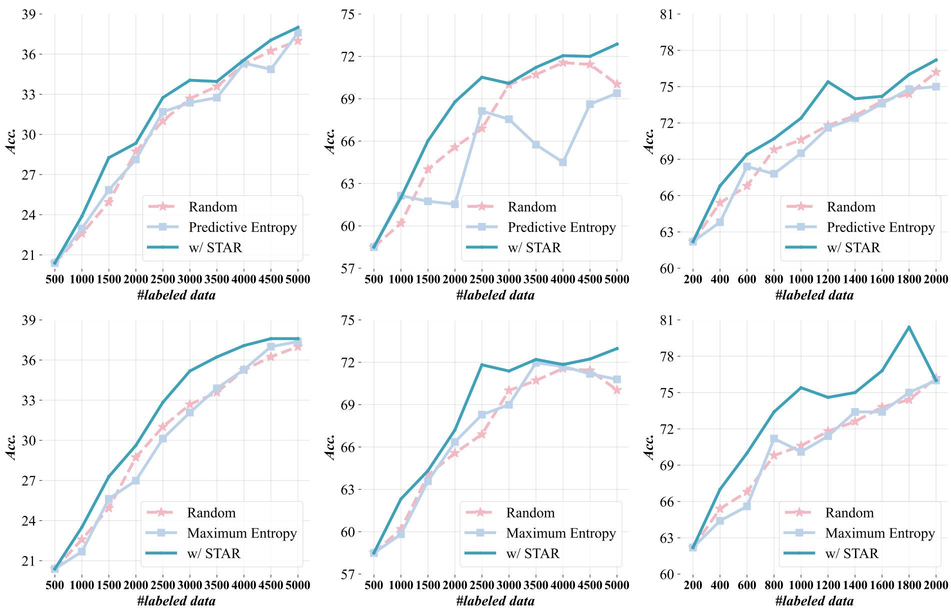

Figure 5: The Learning curves comparing the PREDICTIVE ENTROPY and MAXIMUM ENTROPY methods, and each w/ **STAR**, against the RANDOM baseline. The first column corresponds to the GSM8K dataset, the second column to the BoolQ dataset, and the third column to the OpenBoolQA dataset.

dataset (Cobbe et al., 2021) comprises approximately 8.5K high-quality linguistically diverse grade school math word problems created by human problem writers.

Commonsense Reasoning: (1) the BoolQ dataset (Clark et al., 2019) is a specialized question-answering dataset designed for yes/no questions; (2) the OpenBookQA dataset (Mihaylov et al., 2018) is a four-way multiple-choice question-answering dataset.

See Appendix A for more details about the dataset.

6.2 Settings

Experimental setup In the experiment conducted on the GSM8K and BoolQ datasets, we incrementally selected 500 new instances in each step of the AL experiment. The initial warm start for the AL setting is established by randomly choosing 500 instances. Furthermore, we adhere to a labeling budget constraint of 5,000 instances for each dataset. Considering the size of the training set for OpenBookQA, we design the AL framework to incrementally select 200 new instances during each iteration. The labeling budget for this process is set to 2,000 instances. The details regarding the evaluation can be found in Appendix C.

Implementations In the empirical study, we utilize the state-of-the-art openly accessible LLM, LLaMA2-7B (Touvron et al., 2023b)[2] as the base model. For comprehensive details on the hyperparameters employed in our experiments, please refer to Appendix B.

7 Result and analysis

7.1 Main Result

Table 1 presents a detailed comparison of different methods' performance, evaluated across three

[2] https://huggingface.co/meta-llama/Llama-2-7b

Method	GSM8K		BoolQ		OpenBookQA	
	AUC	RIPL	AUC	RIPL	AUC	RIPL
Predictive Entropy	27.30	-0.09	58.39	-5.24	63.05	-1.07
+Dynamic	27.73	0.59	60.80	0.86	63.81	1.01
+Monte-Carlo dropout	28.04	1.02	61.30	2.12	64.16	1.97
+ L^2 norm weight decay	27.93	0.87	60.99	1.34	64.13	1.89
Maximum Entropy	27.16	-0.28	60.65	0.48	63.36	-0.22
+Dynamic	27.68	0.52	61.01	1.39	64.51	2.95
+Monte-Carlo dropout	28.03	1.00	61.72	3.19	65.17	4.73
+ L^2 norm weight decay	27.82	0.72	61.29	2.10	64.72	3.50

Table 2: The ablation performance of different methods, AUC and RIPL are reported.

Algorithm 1 STAR

Input:unlabeled dataset D^U, labeled dataset D^L, the LLM M, number of iteration N, size of subdataset during iteration m.

initialize D_0^U and D_0^U
warm-up LLM M_0
for $k = 0$ **to** N:
    making inference with M_k on D_k^U
    querying subset S_k^U from D_k^U based on Equation (2)
    updating $D_{k+1}^U \leftarrow D_k^U \setminus S_k^U$
    labeling S_k^U to get S_k^L
    updating $D_{k+1}^L \leftarrow D_k^L \cup S_k^L$
    fine-tuning LLM M_k to get M_{k+1} on D_{k+1}^L based on Equation (6) and Equation (7)
return LLM after fine-tuning M_N

different datasets: GSM8K, BoolQ, and OpenBookQA. Random serves as a fundamental baseline, with its AUC listed.

Both original PE and ME methods underperform compared to Random on these three datasets in terms of AUC. The RIPL metric also hovers around zero, indicating that the original AL strategy is essentially ineffective.

After applying our proposed **STAR** method, PE and ME exhibit superior performance across all datasets and metrics. For instance, in the GSM8k dataset, ME w/ STAR achieves an AUC of **28.83** and a RIPL of **2.01**, indicating a notable advancement over the baseline Random and ME. The improvements are most pronounced in the OpenBookQA dataset, where ME w/ STAR method achieves a remarkable RIPL of **7.47**. Furthermore, in the BoolQ dataset, ME w/ STAR achieves higher performance compared to the PE w/ STAR. This pattern of ME w/ STAR outperforming PE w/ STAR is consistent across the GSM8K and OpenBookQA datasets as well. These results suggest that the ME w/ STAR is more effective.

Then, we explore how the models' performance changes as the training set increases. Figure 5 shows the learning curves for corresponding AL methods on GSM8K, BoolQ, and OpenBookQA datasets, respectively. The Random baseline and the two original active learning approaches perform comparably, suggesting that the active learning methods appear to be ineffective. Notably, the BoolQ dataset exhibits particularly high variability in results when using the PE strategy, which may be attributed to BoolQ's binary output format of "*true*" and "*false*". The gap between the full model and the base model could easily lead to skewed predicting results in a single iteration.

It is evident that w/ STAR methods demonstrate the most significant improvement on the OpenBookQA dataset. After applying our method, the model learned truly more useful samples. For instance, as evidenced by the BoolQ dataset in Figure 5, the performance of the model reaches saturation with just 1000 samples. This indicates that the selected samples are sufficiently diverse and useful for model learning.

7.2 Ablation Study

Since our methods have two main components, which are dynamic uncertain measurement and calibration with hybrid regularization as described in Sec 5. We conduct a detailed ablation study to assess the effect of the two components. As shown in Table 2, upon employing the dynamic uncertain measurement, all AUC are improved, and the RIPL turns positive. This indicates a significant gap between the full model and the base model in the original strategies, which our dynamic indicator effectively mitigated.

Subsequently, building on this foundation and individually incorporating MC dropout and L^2 norm weight decay, it is observed that both constraint methods enhance performance, with MC dropout offering a more substantial improvement. The addition of calibration methods indeed effectively mitigates the issue of model over-confidence and improves model calibration.

8 Conclusion

In this paper, to improve the data efficiency of Large Language Models (LLMs) during the fine-tuning process, we propose a data-efficient parameter tuning method by combining LoRA with active learning. To address the issue that uncertainty-based active learning fails to combine with LoRA, we experimentally identify and summarize two possible reasons: uncertainty gap and poor model calibration. To resolve the uncertainty gap issue, we propose a dynamic uncertainty calculation method, and to address poor model calibration, we introduce a regularization-based constraint method. By integrating these two approaches, we partially solve the aforementioned failure issues. Extensive experiments show that our proposed method outperforms baseline models on multiple reasoning datasets.

Limitations

Though achieving promising results in the experiments, our work still has the following limitations.

- Due to constraints on computational resources, we did not conduct experiments on larger versions of LLaMA2 from 13B to 70B, nor did we experiment with other types of LLMs including BLOOM, Falcon, *etc*.

- Due to limitations in computational resources and time, we did not explore the combination of other types of PEFT methods (series/parallel adapters, prefix tuning) with different types of active learning methods (diversity-based active learning). Therefore, the validity of the methods and conclusions in this paper for a wider combination of PEFT and active learning remains unexplored. Further work should include exploring a more extensive combination of PEFT and active learning.

- We only speculated on the reasons for the failure of combining LoRA with active learning through simple probe experiments, without delving deeper into the underlying mechanisms. Future work should involve exploring the deeper mechanisms behind this phenomenon.

Acknowledgement

The authors would like to thank the anonymous reviewers for their insightful comments. This work is funded by the National Natural Science Foundation of China (62176053). This work is supported by the Big Data Computing Center of Southeast University.

References

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. 2021. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 7319–7328, Online. Association for Computational Linguistics.

Xuefeng Bai, Jialong Wu, Yulong Chen, Zhongqing Wang, and Yue Zhang. 2023. Constituency parsing using llms. arXiv preprint arXiv:2310.19462.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, S. Buch, Dallas Card, Rodrigo Castellon, Niladri S. Chatterji, Annie S. Chen, Kathleen A. Creel, Jared Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren E. Gillespie, Karan Goel, Noah D. Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas F. Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, O. Khattab, Pang Wei Koh, Mark S. Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir P. Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Benjamin Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, J. F. Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Robert Reich, Hongyu Ren, Frieda Rong, Yusuf H. Roohani, Camilo Ruiz, Jack Ryan, Christopher R'e, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishna Parasuram Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei A. Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2021. On the opportunities and risks of foundation models. ArXiv.

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.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168.

David A Cohn, Zoubin Ghahramani, and Michael I Jordan. 1996. Active learning with statistical models. Journal of artificial intelligence research, 4:129–145.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. arXiv preprint arXiv:2305.14314.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shizhe Diao, Pengcheng Wang, Yong Lin, and Tong Zhang. 2023. Active prompting with chain-of-thought for large language models. arXiv preprint arXiv:2302.12246.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. Nature Machine Intelligence, 5(3):220–235.

Chi Dung Doan and Shie-yui Liong. 2004. Generalization for multilayer neural network bayesian regularization or early stopping. In Proceedings of Asia Pacific association of hydrology and water resources 2nd conference, pages 5–8.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey for in-context learning. arXiv preprint arXiv:2301.00234.

Jinhao Duan, Hao Cheng, Shiqi Wang, Chenan Wang, Alex Zavalny, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2023. Shifting attention to relevance: Towards the uncertainty estimation of large language models. arXiv preprint arXiv:2307.01379.

Liat Ein-Dor, Alon Halfon, Ariel Gera, Eyal Shnarch, Lena Dankin, Leshem Choshen, Marina Danilevsky, Ranit Aharonov, Yoav Katz, and Noam Slonim. 2020. Active Learning for BERT: An Empirical Study. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7949–7962, Online. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In international conference on machine learning, pages 1050–1059. PMLR.

Mudasir A Ganaie, Minghui Hu, AK Malik, M Tanveer, and PN Suganthan. 2022. Ensemble deep learning: A review. Engineering Applications of Artificial Intelligence, 115:105151.

Daniel Gissin and Shai Shalev-Shwartz. 2019. Discriminative active learning. arXiv preprint arXiv:1907.06347.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2021. Towards a unified view of parameter-efficient transfer learning. In International Conference on Learning Representations.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. arXiv preprint arXiv:2009.03300.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In International Conference on Machine Learning, pages 2790–2799. PMLR.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In International Conference on Learning Representations.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. 2023. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In Proceedings of the

2023 Conference on Empirical Methods in Natural Language Processing, pages 5254–5276, Singapore. Association for Computational Linguistics.

Berivan Isik, Natalia Ponomareva, Hussein Hazimeh, Dimitris Paparas, Sergei Vassilvitskii, and Sanmi Koyejo. 2024. Scaling laws for downstream task performance of large language models. arXiv preprint arXiv:2402.04177.

Josip Jukić and Jan Snajder. 2023a. Parameter-efficient language model tuning with active learning in low-resource settings. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 5061–5074, Singapore. Association for Computational Linguistics.

Josip Jukić and Jan Snajder. 2023b. Smooth sailing: Improving active learning for pre-trained language models with representation smoothness analysis. In Proceedings of the 2023 CLASP Conference on Learning with Small Data (LSD), pages 11–24, Gothenburg, Sweden. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. arXiv preprint arXiv:2207.05221.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

David D Lewis. 1995. A sequential algorithm for training text classifiers: Corrigendum and additional data. In Acm Sigir Forum, volume 29, pages 13–19. ACM New York, NY, USA.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 4582–4597, Online. Association for Computational Linguistics.

Yixiao Li, Yifan Yu, Chen Liang, Pengcheng He, Nikos Karampatziakis, Weizhu Chen, and Tuo Zhao. 2023. Loftq: Lora-fine-tuning-aware quantization for large language models. arXiv preprint arXiv:2310.08659.

Baohao Liao, Shaomu Tan, and Christof Monz. 2023. Make pre-trained model reversible: From parameter to memory efficient fine-tuning. In Thirty-seventh Conference on Neural Information Processing Systems.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2023a. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. arXiv preprint arXiv:2312.15685.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 61–68, Dublin, Ireland. Association for Computational Linguistics.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2023b. Gpt understands, too. AI Open.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In International Conference on Learning Representations.

Katerina Margatina, Loic Barrault, and Nikolaos Aletras. 2022. On the importance of effectively adapting pretrained language models for active learning. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 825–836, Dublin, Ireland. Association for Computational Linguistics.

Katerina Margatina, Timo Schick, Nikolaos Aletras, and Jane Dwivedi-Yu. 2023. Active learning principles for in-context learning with large language models. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 5011–5034, Singapore. Association for Computational Linguistics.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

Akiva Miura, Graham Neubig, Michael Paul, and Satoshi Nakamura. 2016. Selecting syntactic, non-redundant segments in active learning for machine translation. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 20–29, San Diego, California. Association for Computational Linguistics.

OpenAI. 2022. Introducing ChatGPT.

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

Subhro Roy and Dan Roth. 2016. Solving general arithmetic word problems. arXiv preprint arXiv:1608.01413.

Claudio Filipi Gonçalves Dos Santos and João Paulo Papa. 2022. Avoiding overfitting: A survey on regularization methods for convolutional neural networks. ACM Computing Surveys (CSUR), 54(10s):1–25.

Christopher Schröder, Lydia Müller, Andreas Niekler, and Martin Potthast. 2023. Small-text: Active learning for text classification in python. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, pages 84–95, Dubrovnik, Croatia. Association for Computational Linguistics.

Christopher Schröder, Andreas Niekler, and Martin Potthast. 2022. Revisiting uncertainty-based query strategies for active learning with transformers. In Findings of the Association for Computational Linguistics: ACL 2022, pages 2194–2203, Dublin, Ireland. Association for Computational Linguistics.

Ozan Sener and Silvio Savarese. 2018. Active learning for convolutional neural networks: A core-set approach. In International Conference on Learning Representations.

Burr Settles. 2009. Active learning literature survey.

Yanyao Shen, Hyokun Yun, Zachary Lipton, Yakov Kronrod, and Animashree Anandkumar. 2017. Deep active learning for named entity recognition. In Proceedings of the 2nd Workshop on Representation Learning for NLP, pages 252–256, Vancouver, Canada. Association for Computational Linguistics.

Ard Snijders, Douwe Kiela, and Katerina Margatina. 2023. Investigating multi-source active learning for natural language inference. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2187–2209, Dubrovnik, Croatia. Association for Computational Linguistics.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.

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

Zhongwei Wan, Xin Wang, Che Liu, Samiul Alam, Yu Zheng, Zhongnan Qu, Shen Yan, Yi Zhu, Quanlu Zhang, Mosharaf Chowdhury, et al. 2023. Efficient large language models: A survey. arXiv preprint arXiv:2312.03863, 1.

Xi Wang, Laurence Aitchison, and Maja Rudolph. 2023. Lora ensembles for large language model fine-tuning. arXiv preprint arXiv:2310.00035.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In International Conference on Learning Representations.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35:24824–24837.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38–45, Online. Association for Computational Linguistics.

Ruixuan Xiao, Yiwen Dong, Junbo Zhao, Runze Wu, Minmin Lin, Gang Chen, and Haobo Wang. 2023. Freeal: Towards human-free active learning in the era of large language models. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 14520–14535.

Yuhui Xu, Lingxi Xie, Xiaotao Gu, Xin Chen, Heng Chang, Hengheng Zhang, Zhensu Chen, Xiaopeng Zhang, and Qi Tian. 2023. Qa-lora: Quantization-aware low-rank adaptation of large language models. arXiv preprint arXiv:2309.14717.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Chuanqi Tan, and Chang Zhou. 2023. Scaling relationship on learning mathematical reasoning with large language models. arXiv preprint arXiv:2308.01825.

Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. 2023a. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. arXiv preprint arXiv:2303.16199.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023b. Llmaaa: Making large language models as active annotators. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 13088–13103.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. arXiv preprint arXiv:2303.18223.

Yuekai Zhao, Haoran Zhang, Shuchang Zhou, and Zhihua Zhang. 2020. Active learning approaches to enhancing neural machine translation. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1796–1806, Online. Association for Computational Linguistics.

Dataset	#train	#test	Answer Format
GSM8K	7,473	1,319	Number
BoolQ	9,427	3,270	Letter
OpenBookQA	4,957	500	Letter

Table 3: Details of datasets being evaluated.

A Dataset

Table 3 shows the statistics of the dataset. In light of the unique tasks associated with each dataset, we implement a structured template approach. This template tailors the content and responses to the specificities of each dataset. We give the data templates for each dataset used to fine-tune LLM in Table 4.

B Model Hyper Parameters

Following the prior works (Hu et al., 2021; Li et al., 2023), we maintain the original weights of the backbone architecture unchanged and integrate low-rank adapters into the Multi-Head Attention(MHA) and Feed-Forward Network(FFN) components of all layers. These low-rank adapters are configured with a rank of 64 and a factor of α set to 16, alongside a dropout rate of 0.1 to mitigate overfitting. The model parameters are optimized by AdamW (Loshchilov and Hutter, 2018). We use a batch size of 8 and a learning rate of 1.5e-4 for the GSM8K task and a batch size of 32 and a learning rate of 3e-5 for the BoolQ task and the OBQA task. In the AL setting, the model is trained for a fixed number of epochs: 3 epochs for the GSM8K task, and 15 epochs for both the BoolQ and OBQA tasks. All reported results are averaged over three runs. Our implementation leverages the *PyTorch*[3] framework and *HuggingFace Transformers*[4] library (Wolf et al., 2020). Our experiments are carried out with an NVIDIA A100 80GB GPU.

C Evaluation

Following previous work (Schröder et al., 2022; Jukić and Snajder, 2023b,a), our study utilizes the Area Under the Curve (AUC) metric to assess the comprehensive efficacy of the methods we propose. The accuracy metric (*Acc.*) is employed for evaluating the effectiveness at each individual AL step.

To ascertain the success of AL, we compute the Relative Improvement over Passive Learning

[3] https://github.com/pytorch/pytorch
[4] https://github.com/huggingface/transformers

Dataset	Fine-tuning Data Template
GSM8K	[QUESTION] Answer the above question. First, think step by step and then answer the final number. [ANSWER]
BoolQ	[QUESTION] The correct answer is [ANSWER]
OpenBookQA	[QUESTION] Answer1: [ANSWER_1] Answer2: [ANSWER_2] Answer3: [ANSWER_3] Answer4: [ANSWER_4] The correct answer is [ANSWER]

Table 4: The fine-tuning data template for each dataset.

(RIPL), delineated as follows:

$$RIPL(S_{AL}, S_{PL}) = \frac{AUC(S_{AL}) - AUC(S_{PL})}{1 - AUC(S_{PL})} \quad (8)$$

where S_{AL} and S_{PL} denotes AL methods and RANDOM method. RIPL serves as an estimator for the quotient of the maximal attainable enhancement that an AL approach can secure over the conventional passive learning benchmark. A RIPL score of 1 signifies the epitome of theoretical enhancement, equating to achieving an *Acc.* of 1 during the initial sampling phase and maintaining this optimum performance across all subsequent stages. In contrast, a RIPL score below 0 suggests that the AL strategy is outperformed by passive learning approaches.