context of LLMs is non-trivial, partly due to the billion parameter scale, black-box nature of LLMs, and partly due to the internal reasoning (typically a combination of multiple complex nonlinear functions) being in a different representation space from textual CoT reasoning [2]. We utilize faithfulness metrics proposed in Lanham et al. [11] that quantify the faithfulness of CoT reasoning from LLMs. Specifically, we employ the *Early Answering* strategy, which evaluates the faithfulness of a CoT by sequentially adding each CoT step to the question and querying the LLM for its answer, conditioned on the truncated set of CoT steps. If the answer from the LLM converges towards the final answer as it encounters more CoT steps, it indicates that the CoT explanation is guiding the answer and is more likely to be faithful.

To evaluate the faithfulness of a CoT E shown in Fig. 2a, the early answering strategy involves providing different truncated versions of E and analyzing how the LLM responds to it. For example, if we provide just the first step of E, *i.e.,* Prompt = *"5! equals what ? 1: 5! = 1×2×3×4×5."* and the LLM does not return 120, but it returns 120 when provided with all the steps in E, *i.e.,* Prompt = *"5! equals what ? Step 1: 5! = 1×2×3×4×5. Step 2: 1×2×3×4×5 = 120. Step 3: So the final answer is 120"*, then we can conclude that E is likely to be faithful. Finally, faithfulness is quantified by the area over the curve (AOC) of explanation fraction vs. the percentage of answers consistent with a full explanation. Note that Lanham et al. [11] measures the faithfulness of CoT reasoning at a dataset level. In contrast, we measure faithfulness of each CoT reasoning using probability scores rather than binary correct or incorrect assessments. Following [11], faithfulness is quantified by the area over the curve (AOC) of explanation fraction vs. probability of final answer consistent with a full explanation as shown in Fig. 2b.

3 Eliciting Faithful Reasoning from LLMs

Next, we describe three strategies to improve the faithfulness of CoT reasoning generated by LLMs focusing on different aspects (data, weight, activations) of an LLM, *i.e.,* in-context examples (Sec. 3.1), fine-tuning weights (Sec. 3.2), and activation editing (Sec. 3.3).

3.1 Faithful Reasoning via In-Context Learning

In contrast to traditional ML approaches that require explicit training or fine-tuning on task-specific data, In-Context Learning (ICL) allows an LLM to generalize and adapt its knowledge by learning patterns from a limited set of demonstrations added within the prompt during inference. ICL is a computationally efficient technique that shows an LLM's capability to transfer knowledge to novel tasks without additional parameter updates and can be used for both open- and closed-source LLMs.

In order to improve the faithfulness of CoT reasoning using ICL, we include demonstrations of faithful CoT in-context before the question. The intuition is that each 'faithful' explanation constitutes a set of logical reasoning blocks expressed in natural language, and steering LLMs towards using these filtered faithful reasoning to construct CoT to arrive at an answer, in turn makes their reasoning more faithful. In particular, we consider N in-context examples, each represented as a triple (Q_i, E_i, A_i) for $1 \leq i \leq N$, where Q_i and A_i represents the question and answer associated with the i-th example, while E_i denotes a 'faithful' CoT reasoning for the question Q_i and answer A_i. Mathematically, we can express the set of N in-context examples as $\{(Q_1, E_1, A_1), (Q_2, E_2, A_2), \ldots, (Q_N, E_N, A_N)\}$.

For a given question Q, a language model $\mathcal{F}$ and system prompt S to generate CoT reasoning A_e along with an answer A, the model $\mathcal{F}$ operates as follows, $\mathcal{F} : (Q + S) \rightarrow (A_e + A)$, whereas in-context learning involves passing in the examples as:

$$\mathcal{F}((Q_1, A_1, E_1) + (Q_2, A_2, E_2), \ldots, +(Q_N, A_N, E_N) + Q + S) = A_e + A,$$

where N demonstrations chosen for ICL impact both the accuracy and faithfulness of answers and CoT reasoning. In order to systematically assess the influence of the specific ICL examples chosen, we propose the following sampling strategies.

1) **Deterministic Uniform (DU).** Here, we query the LLM deterministically with temperature $\tau = 0$ to yield (Q, E, A) triplets over the full training set. We then uniformly sample N demonstrations for ICL. Mathematically, this can be expressed as $S(\tau{=}0,\ \texttt{nshot}{=}N,\ \texttt{mode}{=}\text{'uniform'})$ (see Sec. 2).

2) **Deterministic Faithful (DF).** As above, except we select the N most faithful CoT reasoning across the (Q, E, A) triplets, expressed as $S(\tau{=}0,\ \texttt{nshot}{=}N,\ \texttt{mode}{=}\text{'faithful'})$.

3) **Stochastic Uniform (SU).** With this approach, we introduce diversity in eliciting CoT reasoning by sampling at $\tau > 0$, generating 10 samples per question and retaining only the most faithful sample. We then uniformly sample N demonstrations for ICL, expressed as $S(\tau > 0,\ \texttt{nshot}{=}N,\ \texttt{mode}{=}\text{'uniform'})$.

4) **Stochastic Faithful (SF).** Here, we combine stochastic sampling with most faithful selection and select the N most faithful demonstrations for ICL, expressed as $S(\tau > 0,\ \texttt{nshot}{=}N,\ \texttt{mode}{=}\text{'faithful'})$.

Note that we use these strategies in our empirical analysis and use a superscript ^{c} notation to indicate that only (Q, E, A) triplets with correct answers are used, *e.g.,* SF^c indicates that we stochastically generate CoT reasoning, and select the N most faithful triplets that yielded correct answers.

3.2 Faithful Reasoning via Fine-Tuning

Recent progress in LLMs has led to a paradigm shift from the traditional development of models from scratch to an adoption of shared pre-trained LLMs, *e.g.,* BERT [6], GPT [4], Llama [1], that can readily be fine-tuned for specific downstream applications. We utilize a combination of recent techniques like Parameter-Efficient Fine-Tuning (PEFT) [19] and Low-Rank Adaptation (LoRA) [8] that allows efficient fine-tuning LLMs on smaller datasets and reduces the number of trainable parameters by learning low-rank adaptation matrices, making the fine-tuning process more memory and computationally efficient while retaining information that is important for downstream performance.

Our exploration of faithful CoT reasoning via fine-tuning is motivated by Liu et al. [15], Ding et al. [7] which argues that few-shot PEFT are more effective and cost-efficient as compared to ICL. Hence, we investigate the possible benefits of fine-tuning techniques to elicit more faithful CoT reasoning from LLMs. Our study explores a series of selection strategies aimed at enhancing the faithfulness of CoT reasoning. To this end, we curate a variety of datasets for fine-tuning state-of-the-art LLMs with the goal of fine-tuning LLMs with different question, answer, and CoT reasoning examples and understanding their effects on the faithfulness of CoT reasoning generated by the LLM for test samples during inference. In particular, the strategies we employ for the selection of (Q, E, A) triplets used in finetuning are directly analogous to their ICL counterparts described in Sec. 3.1:

1) **Deterministic Uniform (DU).** Selecting all examples (instead of N random examples) for the finetuning dataset: $S(\tau{=}0,\ \texttt{p}{=}100\%,\ \texttt{mode}{=}\text{'uniform'})$.

2) **Deterministic Faithful (DF).** Selecting a percentage of the most faithful examples (instead of the top N) for finetuning: $S(\tau{=}0,\ \texttt{p} < 100\%,\ \texttt{mode}{=}\text{'faithful'})$.

3) **Stochastic Uniform (SU).** Selecting all examples (instead of N random examples) for the finetuning dataset: $S(\tau > 0,\ \texttt{p}{=}100\%,\ \texttt{mode}{=}\text{'uniform'})$.

4) **Stochastic Faithful (SF).** Selecting a percentage of the most faithful examples (instead of the top N) for finetuning: $S(\tau > 0,\ \texttt{p} < 100\%,\ \texttt{mode}{=}\text{'faithful'})$.

As in Sec. 3.1, the superscript ^{c} notation in the empirical analysis indicates that only (Q, E, A) triplets with correct answers were used for fine-tuning.

3.3 Faithful Reasoning via Activation Editing

Seminal works in explainable artificial intelligence have shown that probing analysis [3] can find vectors in the activation space of deep neural networks that correlate to specific properties learned by the underlying model. Formally, editing activations to steer a LLM's behavior involve two key steps - a probing analysis step to identify which components of the model to intervene on, and an editing step which manipulates the activations at run-time. These two steps are detailed below.

Step 1: Probing for Faithfulness. Analyzing a model's internal structures, such as individual neurons or specific mechanisms like convolution or attention, can offer insights into the inner workings of LLMs [12]. A standard tool to understand a model's inner workings is a "*probe*" [3]. Probes are linear classifiers trained on a model's intermediate activations to predict a property like factual

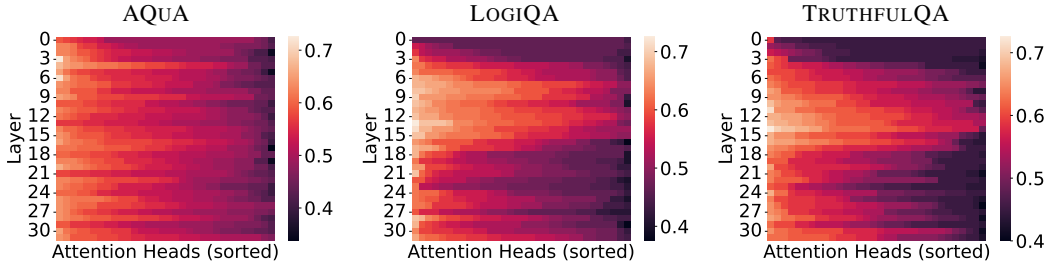

Figure 3: We probe the attention heads across all layers of LLAMA-3-8B-INSTRUCT to assess their predictive power regarding faithfulness. We show the attention heads in each layer sorted by accuracy, clearly indicating that certain attention heads are more responsible for generating faithful explanations.

correctness, harmful biases, etc. By assessing how well these probes perform, we can infer the extent to which certain types of (mis)information is encoded at different layers or components of the model.

Specifically, we aim to identify attention heads that encode information for faithful reasoning. Using a probing dataset of questions, we collect intermediate activations at all layers and attention heads in a LLM, and create a dataset $\{(x_i, y_i)\}_{i=1}^{n}$ for each head h and each layer l, where $x_i \in \mathbb{R}^{d^l_{\text{head}}}$ represents the intermediate activation at a particular layer and attention head of i^{th} question in the probing dataset and y_i represents the faithfulness (measured using approaches described in Sec. 2) of reasoning generated for i^{th} question. The probing dataset is split into 4:1 training and validation sets, and the probe is a logistic regression classifier $\sigma({\theta^l_h}^T \boldsymbol{x})$ to predict faithfulness. As faithfulness is a continuous value, we binarize it using median value as threshold. For a model with L layers and H attention heads, we train a total of $L \times H$ linear probes.

Fig. 3 shows the accuracies of linear probes trained on intermediate activations of LLAMA-3-8B-INSTRUCT on three reasoning and math word problem datasets (discussed at detail in Sec. 4). We observe a significant variance in probing accuracy, suggesting that certain attention heads capture more information about faithful reasoning than others.

Step 2: Activation Editing. Activation editing is a technique to control the post-training behavior of models by using steering intermediate activation vectors, *i.e.,* simple manipulations like translation, scaling, zeroing out, and clamping, on the internal activations of a model at inference time to achieve a desired outcome. By manipulating specific activations associated with certain behaviors, we can alter the LLM's responses without requiring further training. In our exploration, we apply activation editing to improve the faithfulness of CoT reasoning in LLMs. As shown in 3, we first identified specific attention heads that encode more information about faithful CoT reasoning. We then use this information to steer the LLM in the direction that amplifies faithful reasoning. Following Li et al. [12], we translate the activations of a head by a fixed vector during inference.

To avoid causing OoD inputs for subsequent layers by intervening on every head, we do not translate the activations of all attention heads and focus on the top-K heads ranked by the faithfulness metric (Sec. 2), thereby intervening on the LLM's behavior in a minimally invasive manner. The parameters of the linear probe classifier indicate the direction in which faithful and unfaithful reasoning are maximally separable. Thus, we translate in the direction represented by the linear probe parameters θ. where θ^l_h denotes the linear probe classifier trained on the activations on layer l and attention head h

$$\text{Attention}(\mathbf{Q}', \mathbf{K}', \mathbf{V}') = \text{softmax}\left(\frac{\mathbf{Q}'{\mathbf{K}'}^\top}{\sqrt{d_k}}\right)\mathbf{V}' + \alpha\ \theta^l_h\ \sigma^l_h, \tag{1}$$

Figure 4: Attention mechanism used for intervention on attention heads. $\mathbf{Q}'$, $\mathbf{K}'$, and $\mathbf{V}'$ represent query, key, and value matrices respectively. α denotes the intervention strength, θ^l_h represents the learned parameters from linear probe at layer l and attention head h. σ^l_h is a scaling factor.

and α is a hyper-parameter to control the strength of intervention. The direction vector θ^l_h is scaled by σ^l_h, representing the standard deviation of projections of activations in the direction of θ^l_h, ensuring that translation is in the same scale as activations.