B Refusal metric: an efficient proxy for measuring refusal

Evaluating whether a model refuses a particular instruction is most accurately done by generating a full completion using greedy decoding, and then assessing whether the generated text constitutes a refusal. However, this process can be computationally expensive, especially when working with large models and a large number of instructions. To address this, we define a more efficient proxy for estimating the likelihood of a model refusing a given instruction without requiring generation (Figure 10).

We observe that each model tends to have a small set of characteristic phrases that it typically uses to begin its refusals (Figure 9). This allows us to approximate the probability of refusal by examining the model's next token probability distribution at the last token position, which corresponds to the start of its completion.

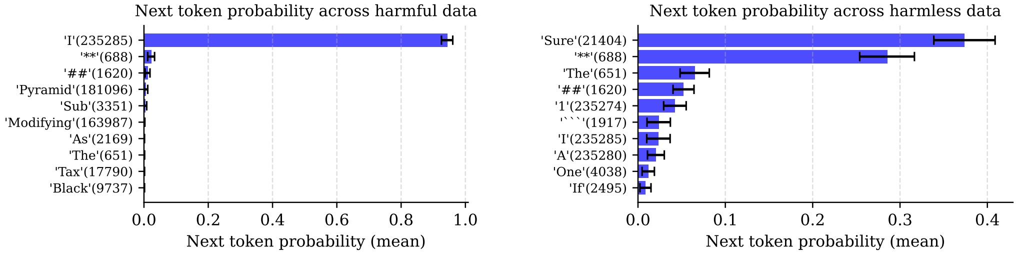
Formally, for each model, we define a set of refusal tokens $\mathcal{R} \subseteq \mathcal{V}$, which contains the tokens most likely to begin the model's refusals (Table 4). We can then estimate the probability of refusal P_{refusal} as the sum of the probabilities assigned to the tokens in $\mathcal{R}$. Given a vector of next token probabilities $\mathbf{p} = (p_1, p_2, \ldots, p_{|\mathcal{V}|}) \in \mathbb{R}^{|\mathcal{V}|}$, we define

$$P_{\text{refusal}}(\mathbf{p}) := \sum_{t \in \mathcal{R}} p_t. \tag{6}$$

To create a more informative "refusal metric", we take the log-odds of P_{refusal}. This transformation helps to better distinguish between extreme probabilities that are close to 0 or 1 (Wang et al., 2023).

$$\begin{aligned}
\texttt{refusal\_metric}(\mathbf{p}) &:= \operatorname{logit}\left(P_{\text{refusal}}(\mathbf{p})\right) && (7)\\
&= \log\left(\frac{P_{\text{refusal}}(\mathbf{p})}{1 - P_{\text{refusal}}(\mathbf{p})}\right) && (8)\\
&= \log\left(\sum_{t \in \mathcal{R}} p_t\right) - \log\left(\sum_{t \in \mathcal{V} \setminus \mathcal{R}} p_t\right). && (9)
\end{aligned}$$

We use this metric to filter out instructions in our test and validation datasets: for harmful instructions, we filter out prompts yielding $\texttt{refusal\_metric} < 0$, and for harmless instruction, we filter out prompts yielding $\texttt{refusal\_metric} > 0$. We also use this metric to quickly evaluate the efficacy of interventions over the validation set (§C).



(a) Top-10 token probabilities for GEMMA 2B IT, averaged over 128 harmful instructions.

(b) Top-10 token probabilities for GEMMA 2B IT, averaged over 128 harmless instructions.

Figure 9: Next token probabilities for GEMMA 2B IT across harmful and harmless instructions.

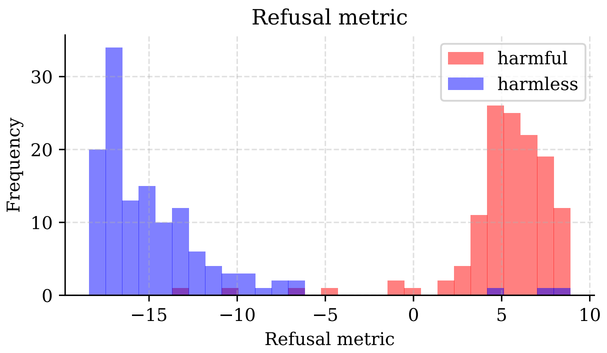


Figure 10: The refusal metric separates harmful and harmless instructions for GEMMA 2B IT. Refusals generally begin with token 234285 (corresponding to '`I`'). Setting $\mathcal{R}_{\text{GEMMA}} = \{234285\}$ yields a refusal metric that is an efficient proxy for assessing whether the model will refuse.

Table 4: The refusal token set $\mathcal{R}$ that we use for each model family, along with the refusal phrases corresponding to each token.

Model family	Refusal token set $\mathcal{R}$	Corresponding refusal phrases
QWEN CHAT	$\{ 40, 2121 \}$	{ `"I'm sorry"`, `"As an AI"` }
GEMMA IT	$\{ 235285 \}$	{ `"I cannot"` }
YI CHAT	$\{ 59597 \}$	{ `"I'm sorry"` }
LLAMA-2 CHAT	$\{ 306 \}$	{ `"I cannot"` }
LLAMA-3 INSTRUCT	$\{ 40 \}$	{ `"I cannot"` }

C Direction selection

C.1 Direction selection algorithm

Given a set of difference-in-means vectors $\{\mathbf{r}_i^{(l)} | i \in I, l \in \lceil L \rceil\}$, we want to select the best vector $\mathbf{r}_{i^*}^{(l^*)}$. For each vector $\mathbf{r}_i^{(l)}$, we compute the following:

- `bypass_score`: under directional ablation of $\mathbf{r}_i^{(l)}$, compute the average refusal metric across $\mathcal{D}_{\text{harmful}}^{(\text{val})}$.
- `induce_score`: under activation addition of $\mathbf{r}_i^{(l)}$, compute the average refusal metric across $\mathcal{D}_{\text{harmless}}^{(\text{val})}$.
- `kl_score`: run the model on $\mathcal{D}_{\text{harmless}}^{(\text{val})}$ with and without directional ablation of $\mathbf{r}_i^{(l)}$, and compute the average KL divergence between the probability distributions at the last token position.

We then select $\mathbf{r}_{i^*}^{(l^*)}$ to be the direction with minimum `bypass_score`, subject to the following conditions:

- `induce_score` > 0
  - This condition filters out directions that are not *sufficient* to induce refusal.
- `kl_score` < 0.1
  - This condition filters out directions that significantly change model behavior on harmless prompts when ablated.
- $l < 0.8L$
  - This condition ensures that the direction is not too close to the unembedding directions. Intuitively, one could disable refusal by preventing the model from writing to refusal unembed directions, e.g. directions corresponding to the `‘I’` or `‘As’` unembedding directions, and this would directly prevent the model from outputting these refusal tokens. However, we restrict our search to higher level features, and do not prevent the model from outputting specific tokens (see §L.1).

Using the compute setup described in §N, this direction selection procedure takes about an hour to run for the largest models (72 billion parameters), and faster for smaller models.

C.2 Direction selection for each model

We report details of direction selection for each model in Table 5, including the token position i^* and layer l^* from which the direction was sourced, along with the direction's corresponding metrics.

Figure 11 displays the `bypass_score` and `induce_score` of all candidate directions for LLAMA-3 8B INSTRUCT.

C.3 Chat templates

We use the default chat template for each model family. All chat templates are displayed in Table 6.