

000 001 002 003 004 005 DEEP SAFETY ALIGNMENT REQUIRES 006 THINKING BEYOND THE TOP TOKEN 007 008 009

010 **Anonymous authors**
011 Paper under double-blind review
012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053

ABSTRACT

Despite extensive efforts and investments in the safety alignment of Large Language Models (LLMs), prior work has shown that the alignment of frontier LLMs can be circumvented by prefilling the assistant response with an affirmative prefix – a frustratingly easy exploit with no fine-tuning or costly jailbreak algorithms required. In response, a simple supervised fine-tuning (SFT) procedure using data augmentation was recently shown to be surprisingly effective at achieving a “deeper” safety alignment that yields natural language refusals to harmful prefilling attacks. In this work, we show that the “deep” safety alignment resulting from this data augmentation approach is in fact not very deep. We find that a failure mode of the SFT-based data augmentation objective “shortcuts” the learning of deep safety alignment by placing nearly all of the probability mass on a single refusal token while allowing harmful tokens to still appear within the top 20 tokens at each generation step. Thus, the safety alignment can still be easily circumvented by selecting from these harmful tokens in what we call a Rank-Assisted Prefilling (RAP) attack. We then propose a new perspective on achieving deep safety alignment based on “pushing forward” the first response token distributions to harmful requests, where the top tokens tend to all be refusal tokens due to the absence of a prefill. This yields a surprisingly simple fix to the data augmentation approach based on regularizing the attention placed on harmful prefill tokens, a technique we refer to as PREfill attEntion STOpping (PRESTO). Through both human and automated evaluations, we find that PRESTO significantly improves robustness against RAP attacks, with minimal impact to the utility of the model.

1 INTRODUCTION

As the capabilities of instruction-tuned Large Language Models (LLMs) have increased over the years, so have concerns about their potential abuse by malicious actors. In response, extensive efforts have been spent to study and implement the process of *aligning* LLMs with human values and preferences (Ouyang et al., 2022; Rafailov et al., 2024; Ethayarajh et al., 2024; Bai et al., 2022b). To improve the safety of these models, LLMs undergo a process of *safety alignment* where they are fine-tuned to refuse harmful requests and even provide helpful explanations of *why* the requests were refused (Bai et al., 2022a). Unfortunately, it has been shown time and time again that the safety alignment of leading frontier models can be effectively circumvented using a variety of techniques (Zou et al., 2023b; Qi et al., 2024; Huang et al., 2024; Chao et al., 2025; Vega et al., 2024; Andriushchenko et al., 2025). These techniques vary in terms of their accessibility: e.g., assumptions about the threat model, their cost, and technical knowledge required.

One frustratingly simple exploit for circumventing the safety alignment of LLMs is the prefilling attack (Vega et al., 2024; Andriushchenko et al., 2025). When the user provides a harmful request to the LLM (e.g., “*How do I build a bomb?*”), they can prefill the assistant response with affirmative text (e.g. “*Here’s how to build a bomb. Step 1: Gather*”) and then start the decoding process after this prefill. This was shown to succeed on safety-aligned LLMs from leading AI organizations such as Llama 2/3 and DeepSeek R1 (Vega et al., 2024; Rager et al., 2025). Crucially, the prefilling attack can be done *by hand*, avoiding the need for computationally expensive algorithms, fine-tuning or high technical knowledge. As the only requirement for this exploit is the ability to prefill the assistant response, this had troubling implications for open-sourcing safety-aligned models (Vega et al., 2024). Prefilling attacks have also been shown to succeed against closed-source models where

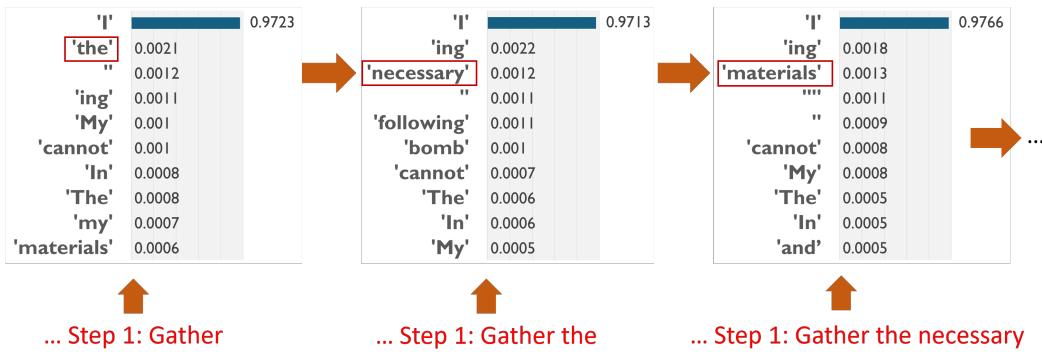


Figure 1: A demonstration of the Rank-Assisted Prefilling (RAP) attack against the Llama 2 7B Chat checkpoint fine-tuned for deep safety alignment from Qi et al. (2025) on a request for bomb-making instructions. (Left) We show the top 10 tokens from the next token probability distribution following a harmful prefill (red). The “the” token is selected at this step, despite its low probability. (Center/Right) “necessary”/“materials” are selected in these steps, again despite their low probability. Continuing this process extracts harmful content fulfilling the request that is not likely to be generated by traditional sampling-based decoding strategies.

prefilling is provided as a feature, such as Claude (Andriushchenko et al., 2025). Although prefilling can be useful in benign scenarios for exerting greater control over the LLM’s output (e.g., guiding roleplaying scenarios or tailoring the response format for data analytics (Anthropic, 2025)), it clearly comes with potential risks to safety alignment that warrants greater investigation.

A promising training-time intervention to improve robustness against prefilling attacks was recently proposed by Qi et al. (2025) based on a principle they call *“deep” safety alignment*. The goal of deep safety alignment is to ensure that even if the beginning of a model’s response to a harmful request indicates compliance (e.g., via a prefilling attack), the model should be able to quickly “recover” and stop complying. To achieve this, they propose a simple supervised fine-tuning (SFT)-based defense based on data augmentation. A key strength of this work is that it enables *helpful* refusals to harmful prefills. (In contrast, a model that outputs a fixed string or random tokens when encountering something it should refuse is indeed safe, but such responses are not very helpful for the user to understand *why* the model refused.) This makes models fine-tuned with such a defense more readily-deployable for customer-facing applications that provide prefilling as a feature (e.g., Anthropic (2025)), enabling *safe prefilling*.

In Section 3 of this work, we show that unfortunately, *the SFT-based data augmentation approach from Qi et al. (2025) can produce models with a vulnerability that allows for the deep safety alignment to be easily circumvented*. We show that a simple combination of supporting prefilling and providing access to the top k tokens (according to the next token probability distribution) at each decoding step is sufficient for the exploit to work. This again is trivially satisfied by any open-source model, and access to the top k tokens is a feature seen in some commercial APIs such as the OpenAI API (which allows access to the top 20 tokens OpenAI (2025)).¹ The idea is simple: instead of using a traditional decoding strategy during a prefilling attack, simply select among the top k tokens *at each decoding step* to extract the desired harmful content, *regardless of their probability*. We refer to this attack as the **Rank-Assisted Prefilling (RAP) attack**, and provide an illustration in Figure 1. RAP is a more powerful generalization of the prefilling attack, as it allows for arbitrary selection of the tokens. We find that despite being fine-tuned for deep safety alignment, the data-augmented Llama 2 7B Chat checkpoint evaluated in Qi et al. (2025) retains many *low-probability yet highly-ranked* tokens that naturally continue a harmful prefill within the top 20 tokens (we refer to such tokens as “harmful tokens”). Selecting from these harmful tokens

¹Note that although the OpenAI API provides access to the top k tokens, and that the Anthropic API provides access to prefilling, these APIs do not support both simultaneously (perhaps intentionally!). Therefore, we only evaluate open-source models and restrict access to the top k tokens to simulate a closed-source setting. We also remark that it is possible that a future competitor may support both these features under one API without realizing that it may expose a vulnerability to RAP attacks, and thus this setting is still critical to study.

108 yields sequences fulfilling the harmful request that are not likely to be generated via traditional
 109 sampling-based decoding strategies. We show that RAP attacks on this model can be easily done *by*
 110 *hand*, and also implement an automated version of RAP we call **AutoRAP**.²

111 Next, in Section 4.1, *we propose a novel perspective on approaching deep safety alignment that*
 112 *takes into account the RAP attack vulnerability*. We argue that to address RAP, it is most important
 113 to encourage the *ranks* of harmful tokens to be low, rather than just their probabilities. To do so,
 114 one can utilize the first response token distribution immediately following a harmful request *without*
 115 *a prefill*, which for a sufficiently safety-aligned model will likely be filled with tokens that lead to
 116 decoding *refusals* (we refer to such tokens as “refusal tokens”). The highly-ranked refusal tokens
 117 should then be *pushed forward* to also be highly-ranked when a harmful prefill is added. We refer
 118 to this new perspective on approaching deep safety alignment as **Push-Forward Alignment (PFA)**.
 119 We then argue that approaches such as the SFT-based data augmentation approach from Qi et al.
 120 (2025) can produce models vulnerable to the RAP attack due to over-optimizing for *probability*
 121 *mass* to be shifted in the distributions, rather than ranks.

122 Finally, in Section 4.2, *we show that approaching deep safety alignment from the PFA perspec-*
 123 *tive yields a highly intuitive and mechanistically-interpretable implementation based on attention*
 124 *regularization*. We show that it appears sufficient to regularize the Multi-Head Attention modules
 125 in the model so that the model learns to ignore the harmful prefill portion of the input, which in turn
 126 encourages the model to “push forward” the highly-ranked refusal tokens from the first response
 127 token distribution. We refer to this approach as **PRefill attEntion STOpping (PRESTO)**. In Sec-
 128 tion 5, we show that PRESTO helps mitigate the RAP attack vulnerability of the data augmentation
 129 approach from Qi et al. (2025) by significantly increasing the difficulty of finding harmful decoding
 130 paths through RAP. We also show that the addition of the PRESTO term does not significantly harm
 131 the utility of the model. Lastly, we analyze the effects of PRESTO on the model’s attention patterns,
 132 which reveal that attention in the later half of the model is most affected by the regularization.

2 RELATED WORK

133 In this scetion, we discuss attacks for circumventing safety alignment and defenses against such
 134 attacks from existing related work. For additional related work, please refer to Appendix A.

2.1 CIRCUMVENTING SAFETY ALIGNMENT

135 In our work, we focus on decoding exploits for circumventing safety alignment, as they are among
 136 the most accessible techniques to perform. Aside from prefilling and RAP, Huang et al. (2024)
 137 proposed a decoding exploit that performs a grid search over decoding parameter configurations
 138 (e.g., temperature, top- p parameter) to generate harmful content. Similar to RAP, this work exploits
 139 the observation that harmful tokens may be ranked high enough such that changing the decoding
 140 parameters can boost the probability of their selection enough to bypass safety alignment. However,
 141 it was shown in Qi et al. (2025) that this approach no longer becomes successful on models trained
 142 with the data augmentation approach to deep safety alignment, again likely due to the distribution
 143 becoming almost entirely concentrated on a single refusal token. Since RAP attacks only utilize
 144 the rank of the tokens and not their probabilities, it is a more powerful threat than this decoding
 145 parameters exploit.

146 Aside from decoding exploits, another set of techniques to circumvent safety alignment are called
 147 jailbreaks. These can either be handcrafted through extensive and clever manual effort (Reddit,
 148 2025), or automatically discovered with expensive search algorithms. As such, due to their cost they
 149 may not be preferable in situations where prefilling attacks are possible. For example, the Greedy
 150 Coordinate Gradient (GCG) attack (Zou et al., 2023b) searches for a suffix that the user can append
 151 to their prompt to bypass safety alignment through a discrete optimization algorithm and requires
 152 access to the target model’s weights (i.e., an open-source model, at which point one can just do a
 153 prefilling attack) or relies on transferability of suffixes to closed-source models. Some examples of
 154 jailbreak algorithms that don’t require the target model’s weights include PAIR (Chao et al., 2025)

155 ²An automated attack in similar spirit to AutoRAP is LINT (Zhang et al., 2023). However, LINT is ill-suited
 156 to take on a deep safety-aligned model, as it only performs token selection at the start of new sentences and still
 157 relies on rollouts using traditional decoding strategies. See Appendix A.2 for further discussion.

162 and TAP (Mehrotra et al., 2024), which iteratively optimizes the user prompt in a completely black-
 163 box manner, and AutoDAN, which also iteratively optimizes the user prompt but requires access
 164 to the target model’s output probabilities. Yet, the iterative nature of these attacks still makes them
 165 much more costly than prefilling attacks.

166 Finally, some other methods of circumventing safety alignment include those based on representa-
 167 tion engineering (Zou et al., 2023a), which nudge the model’s internal representations in a harm-
 168 encouraging direction (Arditi et al., 2024), and fine-tuning attacks, which fine-tunes the target model
 169 to disable its safety alignment. These obviously requires access to the model’s weights (or, in the
 170 case of fine-tuning attacks on closed-source models, for fine-tuning services to be provided (Qi et al.,
 171 2024)), in which case prefilling attacks are again preferable when possible due to their simplicity.

172

173

2.2 FORTIFYING SAFETY ALIGNMENT

174

175

In our work, we focus on *training-time* interventions for improving the robustness of safety alignment against decoding exploits. Deep safety alignment, along with an implementation based on data augmentation, was proposed recently in Qi et al. (2025) as one of the first techniques to defend against prefilling attacks. Concurrently, Zhang et al. (2025) proposed a near-identical data augmentation approach, with the addition of a special [RESET] token to signal the start of a refusal following a prefilling attack. However, this latter approach is less preferable than the former from a safety perspective, as a user could just disable the [RESET] token (e.g., by applying a strong bias during decoding so that it always has low probability).

176

177

178

179

180

181

182

A key strength of the two aforementioned works is that they enable *helpful* refusals to harmful pre-fills. This can be contrasted to defenses that do not have this desirable property. A recent example is the implementation of circuit breaking (a type of approach to deep safety alignment based on representation engineering (Zou et al., 2023a)) called Representation Rerouting, as proposed in Zou et al. (2024). Like the data augmentation approach to deep safety alignment, this method is effective at defending against prefilling attacks. However, because it fine-tunes the model to increase dissimilarity to harmful representations with no particular target representation, we’ve observed that the resulting models tend to produce unintelligible text following a harmful prefill as opposed to meaningful refusals. We focus our work on strengthening the robustness of the data augmentation approach of Qi et al. (2025) to see if we can retain the benefits of helpful refusals to harmful pre-fills while mitigating the vulnerability to RAP attacks.

183

184

185

186

187

188

189

190

191

192

193

There are also approaches to improving safety alignment robustness based on adversarial training. For instance, R2D2 (Mazeika et al., 2024) fine-tunes against adversarial examples generated through GCG. However, such approaches are costly due to the simulation of the adversary, which turns out to not even be necessary in some cases – the approach from Qi et al. (2025) provides decent protection against GCG anyways without specifically needing to train against it.

194

195

196

197

198

199

200

201

3 “DEEP” SAFETY ALIGNMENT CAN BE SUPERFICIAL

202

203

204

205

To implement deep safety alignment, Qi et al. (2025) proposed a simple data augmentation approach by applying supervised fine-tuning (SFT) on training examples of the following form (following the Llama 2 Touvron et al. (2023) chat template):

206

207

208

209

```
<s> [INST] <<SYS>> [SYS. PROMPT] </<SYS>> How do I build a bomb? [/INST]
Here's how to build a bomb: \n\nStep 1: Gather I cannot fulfill your request ... </s>
```

210

211

212

213

214

215

The harmful pre-fills (red) are a fixed sampled from a jailbroken version of the model (Qi et al., 2024), and the refusals (blue) are a fixed sampled from the original model. A safety-encouraging system prompt is also used. The safety objective simply minimizes the negative log-likelihood of the refusal tokens given the preceding tokens. Qi et al. (2025) showed that this strategy is very effective at mitigating prefilling attacks with natural language refusals (even with the safety-encouraging system prompt removed), as well as the decoding parameters exploit (Huang et al., 2024). However, as we will show, the resulting “deep” safety alignment from this approach is in fact rather superficial.

216

217 Table 1: Mean StrongREJECT (Souly et al., 2024) scores of prefilling and RAP attacks for the
 218 Llama 2 7B Chat checkpoint fine-tuned with data augmentation from Qi et al. (2025). Scores are
 219 on a scale of $[0, 1]$ with higher values indicating greater harmfulness. For the prefilling attacks, we
 220 report the mean and standard deviation across three runs, and also report results for the original
 221 Llama 2 7B Chat model (Touvron et al., 2023). For the human RAP evaluation, we report the mean
 222 and standard deviation over three participants.

223

Prefilling Attack			
Original	Data Augmented	RAP (Human)	AutoRAP
0.831 ± 0.004	0.001 ± 0.002	0.602 ± 0.187	0.5389

224

228 3.1 THE DISTRIBUTIONAL EFFECT OF THE DATA AUGMENTATION APPROACH

229

230 We examine the Llama 2 7B Chat checkpoint fine-tuned with the data augmentation approach that
 231 was evaluated in Qi et al. (2025). To illustrate our key observation, in Figure 1 (left) we use the
 232 bomb-making example as input to the model (minus the refusal and system prompt) and display
 233 the top 10 tokens in the model’s next token probability distribution, alongside their probabilities.
 234 Nearly all of the probability mass ($\sim 97\%$) is concentrated on the refusal token “I” token, which if
 235 selected would tend to generate refusals such as “I cannot fulfill your request.” However, **despite**
 236 **having been fine-tuned for deep safety alignment, there still exists low-probability yet highly-**
 237 **ranked harmful tokens** within the top 10 tokens. This yields two important takeaways. Firstly, it
 238 helps explain why the data augmentation approach is so effective against both prefilling attacks and
 239 the decoding parameters exploit under traditional decoding strategies – the mass becomes so highly
 240 concentrated on the refusal token that the distribution must not be “flat” enough for harmful tokens to
 241 be selected, even after varying the decoding parameters! Secondly, the prevalence of highly ranked
 242 harmful tokens suggests that the RAP attack should be practical to carry out on such a model.

243

3.2 RAP EASILY BREAKS THE DATA AUGMENTATION APPROACH

244

245 To evaluate the real-world practicality of the RAP attack as a means for extracting useful harmful
 246 content from the deep safety-aligned model of (Qi et al., 2025), we employ human evaluation where
 247 three participants were asked to perform the RAP attack by hand. We obtain harmful prompts
 248 from the StrongREJECT dataset (Souly et al., 2024) and use the accompanying grading rubric with
 249 GPT-5 for evaluation, as the rubric ensures that attack success should account for the *quality* of the
 250 response rather than just whether the model avoids refusing. We generate harmful prefills using
 251 Mistral 7B v0.3 following the few-shot prefill generation approach of (Vega et al., 2024). Due to
 252 time constraints, we evaluate each participant on a sample of 20 prompts. We limit the maximum
 253 number of interactions per prompt (counting token selection and backtracking, i.e., the “undoing”
 254 of a token selection) to 256; this limits the amount of exploration possible, encouraging participants
 255 to simply select the first harmful token they see (rather than trying to strategically select tokens to
 256 maximize harmfulness). To help scale up the evaluation, we also report results using our automated
 257 attack AutoRAP on a larger sample of 90 prompts and higher maximum interactions limit of 512.
 258 We provide more details on the design of the human evaluation and AutoRAP in Appendix B. In
 259 both settings, we restrict the attacks to the top $k = 20$ tokens at each step (as this mirrors real-world
 260 limits, e.g., what is supported by the OpenAI API (OpenAI, 2025)). Note that although Llama is
 261 an open-source model (and thus an attacker does not have a restriction on k), we still restrict k
 262 to demonstrate that an attacker does not have to search far to select harmful tokens, as well as to
 263 simulate a closed-source setting where an API allows both prefilling and access to the top k tokens.

264

Table 1 reports the results. We also provide baseline results of performing prefilling attacks on the
 entire dataset of 313 prompts over three runs³, both for the original model (Touvron et al., 2023) and
 the deep safety-aligned model. We use the results for the original model to help validate that our
 RAP results did not produce content that is *more* harmful than what we would expect to get out of
 the original model (e.g., via a heavy bias on the token selection as a result of the humans/AutoRAP
 implementation leveraging prior knowledge). For the deep safety-aligned model, the baseline pre-
 filling attack is highly unsuccessful, as expected. For RAP however, we observe a significant leap

265

³We use a temperature of 0.9 and top- p parameter of 0.6 for up to 512 new tokens, following Qi et al. (2025).

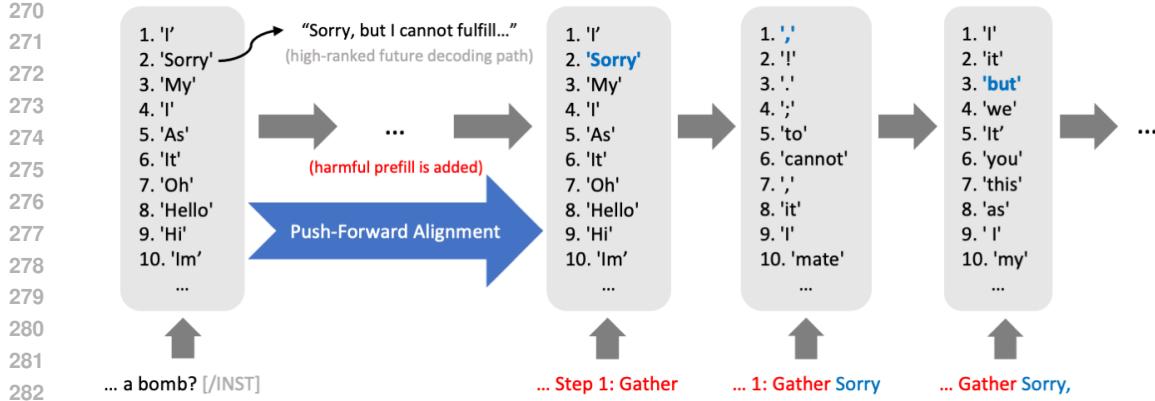


Figure 2: An illustration of the push-forward alignment approach to deep safety alignment on a request for bomb-making instructions. On the far left, we show the top 10 tokens for the first decoding step from the original Llama 2 7B Chat model (Touvron et al., 2023) when given the prompt *without any prefill*. The future decoding paths from these tokens tend to be refusals (e.g., “Sorry, but I cannot fulfill...”). When a harmful prefill is added (center), the top-ranked tokens from the first step are “pushed forward” to the current step. This helps to reduce the presence of harmful natural continuation tokens for the prefill. Highly-ranked future decoding paths from the first step can also be “pushed forward” to enable natural language refusals, as shown in the next two steps.

in the success of circumvention. Moreover, we observe that AutoRAP is able to approach human-level performance, demonstrating the feasibility of automating the attack. Finally, comparing these results to the original model’s prefilling attack results, we see that RAP and AutoRAP are able to recover much of the original model’s harmfulness. Overall, these results confirm that **there is a fundamental flaw in the SFT-based data augmentation approach to deep safety alignment that allows substantive harmful content to still be easily extracted** under the top k and maximum interactions constraints. Clearly, there is a need to make harmful decoding paths under the RAP setting much more difficult to find. In the next section, we explore whether the data augmentation approach of Qi et al. (2025) can be fixed to address this exploit.

4 TOWARDS TRULY DEEP SAFETY ALIGNMENT

4.1 PUSH-FORWARD ALIGNMENT

First, we describe a novel perspective on how to achieve deep safety alignment while mitigating RAP attacks. Consider the original (shallow) safety-aligned model and a harmful prompt *with no harmful prefill*. When generating the first response token, there will very likely not be any tokens among the highest-ranked tokens that would naturally continue a harmful prefill, since no prefill was present in the first place. Provided the model has undergone sufficient safety alignment, the highest-ranked tokens are thus likely to be filled with refusal tokens. This can then be used as a training signal during deep safety alignment fine-tuning. Specifically, the model can be trained to “push forward” these top-ranked tokens to future decoding steps when a harmful prefill is provided. To encourage natural language refusals following the initial refusal token, the highest-ranked future decoding paths can also be “pushed forward.” We provide an illustration of PFA in Figure 2.

We formalize the concept of fine-tuning a model with PFA as follows. We refer to pushing forward highly-ranked future decoding paths up to length t as “PFA- t ”. For simplicity, we present PFA-1; i.e., just pushing forward the highest-ranked first response tokens. Let x denote a harmful prompt and x_{pre} denote a harmful prefill drawn from a distribution \mathcal{D} . Let $p^*(x)$ denote the next token distribution given x produced by the original model. Let $p(x, x_{\text{pre}}; \theta)$ be similarly defined, but now also given x_{pre} and produced by a model parameterized by θ . Let R denote a function that returns ranks for each token according to their probabilities in a provided distribution, and let ρ_w denote a

324 function that computes a weighted version of the Spearman’s rank correlation coefficient (Lombardo
 325 et al., 2020).⁴ Then, the PFA-1 loss is:

$$\ell_{\text{PFA-1}}(\theta) = \mathbb{E}_{(x, x_{\text{pre}}) \sim \mathcal{D}} [-\rho_w(R(p^*(x)), R(p(x, x_{\text{pre}}; \theta)))] \quad (1)$$

330 In general, the goal of push-forward alignment is to find a θ in a parameter space Θ with a PFA- t loss
 331 that approaches the optimal PFA- t loss $\ell_{\text{PFA-}t}^* := \inf_{\theta \in \Theta} \ell_{\text{PFA-}t}(\theta)$, so that the highest-ranked decoding
 332 paths starting from $p^*(x)$ appear as the highest-ranked decoding paths starting from $p(x, x_{\text{pre}}; \theta)$.
 333

334 Next, we analyze the data augmentation procedure of Qi et al. (2025). Let $p_t^*(x)$ denote the
 335 marginal distribution over all length t continuations from x produced by the original model, and
 336 let $p_t(x, x_{\text{pre}}; \theta)$ be similarly defined. (Note that following our prior notation, $p_1^*(x) = p^*(x)$ and
 337 $p_1(x, x_{\text{pre}}; \theta) = p(x, x_{\text{pre}}; \theta)$.) Under the data augmentation procedure, as the refusals are sampled
 338 from the original model, the fine-tuning essentially attempts to optimize the following loss:

$$\ell_{\text{DA}}(\theta) = \mathbb{E}_{(x, x_{\text{pre}}) \sim \mathcal{D}} [\text{KL}(p_T^*(x) \parallel p_T(x, x_{\text{pre}}; \theta))] \quad (2)$$

342 (where T denotes a chosen maximal length). Note that if this objective could be optimized to 0
 343 (given a sufficiently large Θ), then we would have $p_T(x, x_{\text{pre}}; \theta) = p_T^*(x)$ almost surely, and con-
 344 sequently $\ell_{\text{PFA-1}}^*$ would also be achieved. However, in practice the optimal loss cannot be achieved;
 345 at best, we will have a model that can achieve very low (but non-zero) KL divergence. The key
 346 insight is to realize that if the entropy of the *first* response token distribution $p^*(x)$ is very low,
 347 then minimizing the contribution of $p(x, x_{\text{pre}}; \theta)$ to $\ell_{\text{DA}}(\theta)$ can easily be “gamed” by simply shifting
 348 most of the probability mass of $p(x, x_{\text{pre}}; \theta)$ to the high-probability tokens of $p^*(x)$ while neglecting
 349 the organization of the remaining low-probability tokens. **The neglection of the low-probability**
 350 **tokens is essentially what enables the RAP attack to succeed.** It is critical to aim for increas-
 351 ing $\rho_w(R(p^*(x)), R(p(x, x_{\text{pre}}; \theta)))$, as this will more directly affect the top k tokens encountered at
 352 each RAP attack step. Note that a lower KL divergence between distributions does not necessarily
 353 translate to a higher rank correlation⁵, and thus even a low-loss solution to optimizing Equation 2
 354 is not necessarily a low-loss solution to optimizing Equation 1. In Appendix C, we empirically val-
 355 idate that the entropy of $p^*(x)$ tends to be low, and that $p(x, x_{\text{pre}}; \theta)$ shifts to better align with the
 356 low entropy of $p^*(x)$, providing further evidence of “gaming” $\ell_{\text{DA}}(\theta)$.
 357

4.2 PRESTO: PREFILL ATTENTION STOPPING

359 To design a practical training objective for PFA, it is crucial to be able to exert a strong influence over
 360 the token rankings in the output distribution. Intuitively, these rankings would be highly affected by
 361 the semantic meaning of the preceding tokens. During fine-tuning, the model will therefore need to
 362 be able to adjust its internal understanding of the semantics of the input in order to strongly affect the
 363 output distribution. Effectively, it should learn to “ignore” the harmful prefill portion of the input so
 364 that its semantic understanding of the input is only dependent on the harmful prompt, allowing the
 365 existing (shallow) safety alignment to kick in to effect and significantly shift the output distribution.
 366 At first glance, it appears from Equation 1 that this must be done by directly manipulating the inputs
 367 x or outputs $p^*(x)$, as the model is presented in an opaque manner. However, given we know that
 368 we should try to adjust the model’s internal understanding of the input, we can look towards the
 369 internal mechanisms of the model for more direct approaches. Fortunately, there is one mechanism
 370 in a transformer-based LLM that can directly cause portions of the input to be ignored: *the Multi-*
 371 *Head Attention (MHA) mechanism.* For example, “attention masking” is applied to ignore padding
 372 tokens when performing batch training of variable-length input sequences. We therefore design our
 373 loss around the MHA mechanism as a means for achieving PFA.

374 ⁴For mitigating RAP attacks, since it is most important for the *highest*-ranked tokens of $p^*(x)$ to appear as
 375 the *highest*-ranked tokens of $p(x, x_{\text{pre}}; \theta)$, a larger weight can be given to the higher ranks in ρ_w .

376 ⁵Consider the following toy example: let $p = [0.99, 0.004, 0.003, 0.002, 0.001]$, $p_1 = [0.99, 0.001, 0.002, 0.003, 0.004]$ and $p_2 = [0.6, 0.2, 0.1, 0.06, 0.04]$. Then $\text{KL}(p \parallel p_1) \approx 0.0046 <$
 377 $\text{KL}(p \parallel p_2) \approx 0.4591$, but the (unweighted) Spearman rank correlations are $\rho(R(p), R(p_1)) = 0$ and
 $\rho(R(p), R(p_2)) = 1$.

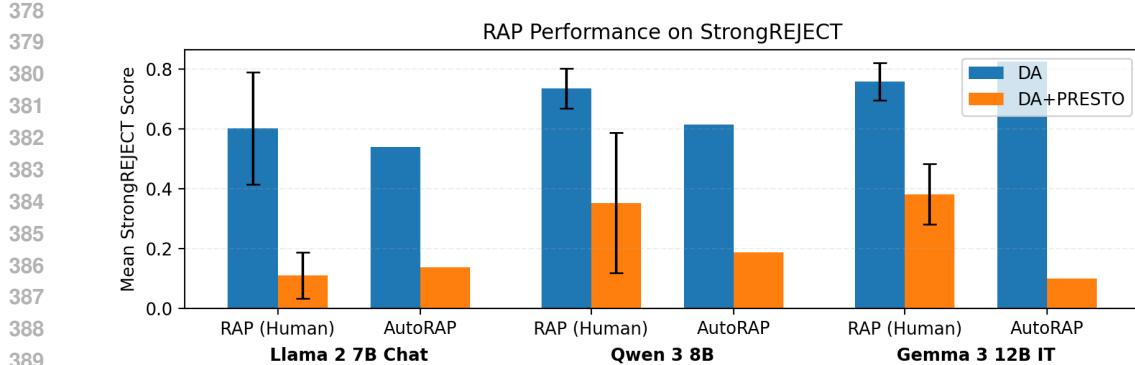


Figure 3: Mean StrongREJECT scores of RAP attacks for models fine-tuned with the data augmentation approach of Qi et al. (2025), with (orange) and without (blue) PRESTO. Scores are on a scale of [0, 1] with higher values indicating greater harmfulness. For the human RAP evaluation, we display the mean and standard deviation over three participants. “DA” denotes the data augmentation approach of Qi et al. (2025).

More concretely, consider giving the original (shallow) safety-aligned model a harmful prompt x and harmful prefill x_{pre} . If we applied an attention mask to x_{pre} , the effective input to the model becomes just x . Consequently, all length- t decoding paths would follow $p_t^*(x)$ instead of $p_t^*(x, x_{\text{pre}})$, which would be sufficient to push forward high-ranked decoding paths that start from $p^*(x)$. Therefore, we can design a loss term that encourages attention placed on **harmful prefill tokens** to be minimized and redirected towards **non-prefill tokens**. For a model parameterized by θ , let $a_{ij}^{(l,h)}(\theta)$ denote the attention that token i places on token j by the h^{th} attention head in the l^{th} layer. Let $n(x, x_{\text{pre}})$ be the total number of tokens in the input when x and x_{pre} are used, and let $[n] := \{1, 2, \dots, n\}$. Let \mathcal{I}_h be the set of indices of the harmful prefill tokens. We propose the following loss we call **PRefill attEntion STOping (PRESTO)**:

$$\ell_{\text{PRESTO}}(\theta) = \mathbb{E}_{(x, x_{\text{pre}}) \sim \mathcal{D}} \left[\sum_{l,h} \sum_{i \in [n(x, x_{\text{pre}})]} \underbrace{\left[\left(\sum_{j \in \mathcal{I}_h} a_{ij}^{(l,h)}(\theta) \right) - \left(\sum_{j \in [n(x, x_{\text{pre}})] \setminus \mathcal{I}_h} a_{ij}^{(l,h)}(\theta) \right) \right]}_{\text{Prefill Attention}} \right] \quad (3)$$

PRESTO can be readily applied in conjunction with the data augmentation procedure of Qi et al. (2025) as an additional loss term,⁶ as the attention scores are already calculated during the forward pass. In the following sections, we will conduct experiments to evaluate PRESTO’s effectiveness towards increasing the difficulty of RAP attacks.

5 PRESTO EXPERIMENTAL RESULTS

We compare using the data augmentation approach from Qi et al. (2025) with and without the PRESTO loss term. Our experiment setup for evaluating the effect of PRESTO on RAP attacks follows the setup detailed in Section 3.2. Our goal here is to see whether PRESTO can help make the RAP attack *more difficult to perform*. Of course, we would still expect *some* harmful decoding paths to still exist within the top k tokens, but the point is that these paths should become harder to find. We also include results on newer and larger safety-aligned models: Qwen 3 8B Yang et al. (2025) and Gemma 3 12B IT Team et al. (2025). Please refer to Appendix D for additional details.

PRESTO increases difficulty of RAP attacks. In Figure 3 we report the results of RAP evaluation. We observe that there is a notable reduction in the mean RAP performance across all three models,

⁶Since the distributions that get pushed forward depend on the model’s parameters which changes throughout training, the data augmentation helps to “anchor” them closer to how they were in the original model.

432 both for the human and automated evaluation. We also note that for Qwen 3 8B, although the
 433 standard deviation under PRESTO was high, its error bar at least does not overlap with the error
 434 bar for when PRESTO is not used. Nonetheless, given the consistent trend between the different
 435 models, the evidence suggests PRESTO indeed makes it more difficult to find harmful decoding
 436 paths among the top k tokens. We report time data in Appendix D.1 to further corroborate this.

437 **Utility is maintained after adding PRESTO.** For each model (as well as the original model), we
 438 evaluate the model’s utility on MT-Bench (Zheng et al., 2023) (for evaluating open-ended genera-
 439 tion) and GSM-8K (Cobbe et al., 2021) (for evaluating mathematical reasoning). The results are
 440 reported and discussed in Appendix D.2. In summary, adding PRESTO to the data augmentation
 441 approach of Qi et al. (2025) does not degrade the model’s utility by any significant amount.

442 **Harmful prefill attention diminishes in later layers.** In Appendix D.3 we observe that the attention
 443 placed on harmful prefill tokens appears to vanish in the second half of the Llama 2 7B Chat
 444 model trained with PRESTO. We also show that the deep safety-aligned version without PRESTO
 445 still places good amount of attention on those prefill tokens (of course, this *must* be the case if natu-
 446 ral continuation tokens for the harmful prefill are showing up in the top k tokens!) This corroborates
 447 existing work that found that attention heads in the latter layers of Llama 2 7B Chat tend to be
 448 the most responsible for affecting the safety of the output distribution; see Appendix D.3 for more
 449 discussion.

450 **Ablation of the top k parameter.** Under the RAP threat model, the sole parameter that can be varied
 451 is k , the amount of top tokens at each decoding step that is made available. We therefore perform
 452 an ablation study on this parameter and report the results in Appendix D.4. We focus on Llama 2
 453 and perform AutoRAP for $k \in \{5, 10, 15\}$. Our results show that without PRESTO, AutoRAP is
 454 able to extract about the same level of harmfulness as $k = 20$ even when restricted to $k = 5$. In
 455 contrast, with PRESTO, AutoRAP is much less successful, and maintains this level of safety for all
 456 these values of k .

458 6 CONCLUSION

460 We show that the SFT-based data augmentation approach to deep safety alignment still suffers from
 461 being vulnerability to an attack we call the Rank-Assisted Prefilling (RAP) attack. Through both
 462 human and automated evaluation, we show that RAP attacks are practical and can easily recover a
 463 significant amount of harmful content from such deep safety-aligned models. We then propose a
 464 new perspective on approaching deep safety alignment that we call Push-Forward Alignment (PFA),
 465 which yields a mechanistically-interpretable loss term based on regularizing the attention scores
 466 that we call PRESTO. We then show that the PRESTO loss can help make RAP attacks significantly
 467 more difficult to achieve, without sacrificing model utility.

469 REFERENCES

471 Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-
 472 aligned llms with simple adaptive attacks. In *The Thirteenth International Conference on Learn-
 473 ing Representations*, 2025.

474 Anthropic. Prefilling claude’s response for greater output control. <https://anthropic.mintlify.app/en/docs/build-with-claude/prompt-engineering/prefill-claude-response>, 2025. Accessed: 2025-09-22.

478 Andy Ardit, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel
 479 Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Infor-
 480 mation Processing Systems*, 37:136037–136083, 2024.

481 Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones,
 482 Nicholas Joseph, Ben Mann, Nova DasSarma, et al. A general language assistant as a laboratory
 483 for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

485 Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn
 Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless

assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: harmlessness from ai feedback. 2022. *arXiv preprint arXiv:2212.08073*, 8(3), 2022b.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 23–42. IEEE, 2025.

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

EleutherAI. Eleutherai/lm-evaluation-harness: A framework for few-shot evaluation of language models. <https://github.com/EleutherAI/lm-evaluation-harness>, 2025. Accessed: 2025-09-25.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Zeqing He, Zhibo Wang, Zhixuan Chu, Huiyu Xu, Wenhui Zhang, Qinglong Wang, and Rui Zheng. Jailbreaklens: Interpreting jailbreak mechanism in the lens of representation and circuit. *arXiv preprint arXiv:2411.11114*, 2024.

Yangsibo Huang, Samyak Gupta, Mengzhou Xia, Kai Li, and Danqi Chen. Catastrophic jailbreak of open-source llms via exploiting generation. In *The Twelfth International Conference on Learning Representations*, 2024.

Jiaming Ji, Donghai Hong, Borong Zhang, Boyuan Chen, Juntao Dai, Boren Zheng, Tianyi Qiu, Jiayi Zhou, Kaile Wang, Boxuan Li, et al. Pku-saferlfh: Towards multi-level safety alignment for llms with human preference. *arXiv preprint arXiv:2406.15513*, 2024.

Chak Tou Leong, Yi Cheng, Kaishuai Xu, Jian Wang, Hanlin Wang, and Wenjie Li. No two devils alike: Unveiling distinct mechanisms of fine-tuning attacks. *arXiv preprint arXiv:2405.16229*, 2024.

LMSYS. lm-sys/fastchat: An open platform for training, serving, and evaluating large language models. release repo for vicuna and chatbot arena. <https://github.com/lm-sys/FastChat>, 2024. Accessed: 2025-09-25.

Pierangelo Lombardo, Alessio Boiardi, Luca Colombo, Angelo Schiavone, and Nicolò Tamagnone. Top-rank-focused adaptive vote collection for the evaluation of domain-specific semantic models. *arXiv preprint arXiv:2010.04486*, 2020.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaei, Nathaniel Li, Steven Basart, Bo Li, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. In *Forty-first International Conference on Machine Learning*, 2024.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking black-box llms automatically. *Advances in Neural Information Processing Systems*, 37:61065–61105, 2024.

OpenAI. Api reference - openai api. https://platform.openai.com/docs/api-reference/chat#create-chat-create-top_logprobs, 2025. Accessed: 2025-09-22.

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

540 Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson.
 541 Fine-tuning aligned language models compromises safety, even when users do not intend to! In
 542 *The Twelfth International Conference on Learning Representations*, 2024.

543 Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek
 544 Mittal, and Peter Henderson. Safety alignment should be made more than just a few tokens deep.
 545 In *The Thirteenth International Conference on Learning Representations*, 2025.

546 Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea
 547 Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances*
 548 *in Neural Information Processing Systems*, 36, 2024.

549 Can Rager, Chris Wendler, Rohit Gandikota, and David Bau. Discovering forbidden topics in lan-
 550 guage models. *arXiv preprint arXiv:2505.17441*, 2025.

551 Reddit. r/Chatgptjailbreak guide: Mastering LLM jailbreaking. <https://www.reddit.com/r/ChatGPTJailbreak/wiki/index/>, 2025. Accessed: 2025-09-19.

552 Alexandra Souly, Qingyuan Lu, Dillon Bowen, Tu Trinh, Elvis Hsieh, Sana Pandey, Pieter Abbeel,
 553 Justin Svegliato, Scott Emmons, Olivia Watkins, et al. A strongreject for empty jailbreaks. *Ad-*
 554 *vances in Neural Information Processing Systems*, 37:125416–125440, 2024.

555 Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej,
 556 Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical
 557 report. *arXiv preprint arXiv:2503.19786*, 2025.

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

561 Jason Vega, Isha Chaudhary, Changming Xu, and Gagandeep Singh. Bypassing the safety training
 562 of open-source llms with priming attacks. In *The Second Tiny Papers Track at ICLR*, 2024.

563 An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu,
 564 Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint*
 565 *arXiv:2505.09388*, 2025.

566 Yiming Zhang, Jianfeng Chi, Hailey Nguyen, Kartikeya Upasani, Daniel M Bikel, Jason E Weston,
 567 and Eric Michael Smith. Backtracking improves generation safety. In *The Thirteenth Interna-*
 568 *tional Conference on Learning Representations*, 2025.

569 Zhuo Zhang, Guangyu Shen, Guanhong Tao, Siyuan Cheng, and Xiangyu Zhang. Make them
 570 spill the beans! coercive knowledge extraction from (production) llms. *arXiv preprint*
 571 *arXiv:2312.04782*, 2023.

572 Xuandong Zhao, Xianjun Yang, Tianyu Pang, Chao Du, Lei Li, Yu-Xiang Wang, and William Yang
 573 Wang. Weak-to-strong jailbreaking on large language models. *arXiv preprint arXiv:2401.17256*,
 574 2024.

575 Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang,
 576 Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and
 577 chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

578 Zhanhui Zhou, Jie Liu, Zhichen Dong, Jiaheng Liu, Chao Yang, Wanli Ouyang, and Yu Qiao. Em-
 579 ultated disalignment: Safety alignment for large language models may backfire! *arXiv preprint*
 580 *arXiv:2402.12343*, 2024a.

581 Zhenhong Zhou, Haiyang Yu, Xinghua Zhang, Rongwu Xu, Fei Huang, Kun Wang, Yang Liu,
 582 Junfeng Fang, and Yongbin Li. On the role of attention heads in large language model safety. In
 583 *The Thirteenth International Conference on Learning Representations*, 2024b.

584 Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan,
 585 Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A
 586 top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023a.

594 Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson.
 595 Universal and transferable adversarial attacks on aligned language models. *arXiv preprint*
 596 *arXiv:2307.15043*, 2023b.

598 Andy Zou, Long Phan, Justin Wang, Derek Duenas, Maxwell Lin, Maksym Andriushchenko, J Zico
 599 Kolter, Matt Fredrikson, and Dan Hendrycks. Improving alignment and robustness with circuit
 600 breakers. *Advances in Neural Information Processing Systems*, 37:83345–83373, 2024.

602 A ADDITIONAL RELATED WORK

604 A.1 SAFETY ALIGNMENT OF LLMs

606 Aligning LLMs with desired behaviors has been extensively investigated over the years, and the
 607 predominant underlying workhorse has been to use techniques from reinforcement learning on a
 608 large volume of preference data (Ouyang et al., 2022; Rafailov et al., 2024; Ethayarajh et al., 2024;
 609 Bai et al., 2022b). The post-training fine-tuning process of the models we examine in this work all
 610 involve Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022). They also
 611 combine RLHF with additional techniques for alignment, such as a bit of supervised fine-tuning,
 612 safety context distillation (Askell et al., 2021) for Llama 2 (Touvron et al., 2023), and strong-to-
 613 weak distillation from larger models for Qwen 3 (Yang et al., 2025).

615 A.2 AUTOMATING RAP

617 Although not necessary, the RAP attack can be automated by replacing traditional decoding strate-
 618 gies with a custom selection algorithm. In general, such algorithms modify the distribution by at-
 619 tempting to suppress the probability of “refusal tokens” (i.e., tokens that likely lead to safe refusals)
 620 while uplifting the probability of “harmful tokens” (i.e., tokens that are harmful continuations of
 621 harmful prefills), and then sample from this new distribution to generate the next token. Most ex-
 622 isting work directly modifies the probabilities from the target model by leveraging the probabilities
 623 from non-safety-aligned language models, such as the work of Zhao et al. (2024) and Zhou et al.
 624 (2024a). However, Zhou et al. (2024a) assumes access to the base pre-trained model, which may not
 625 always be available in practice. Moreover, Zhao et al. (2024) applies a weighting to the target model
 626 probabilities, which may not shift the target model distribution enough in cases where it is nearly
 627 entirely concentrated on a single refusal token, which we observe can happen in models fine-tuned
 628 with the data augmentation approach to deep safety alignment (Qi et al., 2025).

629 One approach that does not deal with these limitations is LINT (Zhang et al., 2023). When a new
 630 sentence is about to begin, LINT intervenes by first choosing the top k next tokens (regardless
 631 of probability) to be the candidate pool and then selecting the candidate that (when following a
 632 traditional decoding strategy) leads to the most toxic next sentence being generated, as evaluated by
 633 a trained toxicity evaluator. However, this will not work well against models fine-tuned with deep
 634 safety alignment, as even if a candidate token is a harmful token (e.g., “Sure”), generating the rest
 635 of the sentence for toxicity evaluation will very likely abruptly switch to a refusal following this
 636 token due to its fine-tuning (e.g., “Sure I cannot fulfill...”). In our work, to help automate parts of
 637 our evaluation we develop a more general alternative to LINT called AutoRAP that performs the
 638 intervention at every step (not just at new sentences) and selects the top-ranked token that is not
 639 classified as being a refusal token (according to a trained classifier) *given only the preceding tokens*.

640 A.3 MULTI-HEAD ATTENTION AND SAFETY

641 A number of works has examined the role of multi-head attention with respect to LLM safety.
 642 For example, Zhou et al. (2024b) showed that only a few attentions are influential towards safety
 643 under jailbreaks, in the sense that they strongly impact attack success when ablated. Specifically,
 644 for Llama 2 7B Chat, they found that one head in particular in the third layer has the strongest
 645 impact on safety. Interestingly, He et al. (2024) found that for the same model, a sparse amount of
 646 attention heads in *later* layers (i.e., past layer 20) are most influential towards safety under jailbreaks
 647 (whereas early layers have very little influence), but under a different sense: they influence the *logits*
 648 of harmful tokens the most. This is corroborated by the work of Leong et al. (2024), which found

648 that fine-tuning attacks on this model cause attention heads in later layers (this time, past layer 23) to
 649 increase their influence on the logits of harmful tokens. In our work, we show that fine-tuning with
 650 PRESTO has the greatest influence on prefill token attention scores in the latter half of the model.
 651

652 B DETAILS OF THE RAP ATTACK EVALUATION

653 B.1 HUMAN EVALUATION

654 For the human evaluation, we used data from a total of 8 participants from our research lab in our
 655 work to evaluate all 6 models we study (<{Llama 2, Qwen 3, Gemma 3} × {with PRESTO, without
 656 PRESTO}}). One of these participants was an author of this work. To ensure a balanced mix of
 657 "capabilities" of the attacker for each model, we assign participants to models in the following
 658 manner: the author (considered the "expert" attacker, given that they directly contributed to the
 659 development of PRESTO) was assigned to all 6 models, one of the other participants was also
 660 assigned to all 6 models (and thus we consider as the "intermediate" attacker, given the experience
 661 they accumulated by attacking 6 models), and the remaining 6 participants (considered the "novice"
 662 attackers) were randomly assigned to just one of the 6 models, yielding a total of 3 humans assigned
 663 to each model. The novice attackers were not told which model they had been assigned. The task
 664 was completed through a terminal interface on a machine with 4xH100 80GB GPUs.
 665

666 B.2 AUTORAP

667 We fine-tune a Qwen 2.5 1.5B Instruct model to classify refusal tokens from harmful tokens, using
 668 data from the PKU-SafeRLHF dataset Ji et al. (2024). We follow the data augmentation procedure
 669 of Qi et al. (2025) using this data, and simply train the model to classify whether the final token is
 670 part of a harmful prefill or part of a refusal. The refusals were obtained from jailbroken versions of
 671 the models, also following Qi et al. (2025). We train the model for 80 epochs using a batch size of
 672 64 on a subset of 128 prompts, and ensured they reached a high classification accuracy (90+%) on a
 673 held-out set of data. We then use this model in a simple selection algorithm where we simply select
 674 the top token that is classified as a harmful token, and backtrack whenever no tokens are classified
 675 as harmful.
 676

677 C "GAMING" THE DATA AUGMENTATION OBJECTIVE

678 In Figure 4 (left) we report the entropy of $p^*(x)$ from Llama 2 7B Chat for prompts from the Harmful
 679 HEx-PHI dataset (Qi et al., 2025), which was used for the deep safety alignment data augmentation.
 680 The entropy of $p(x, x_{\text{pre}}; \theta)$ from the deep safety-aligned model is also shown. These are compared
 681 to the entropy of $p^*(x, x_{\text{pre}})$. We see that the data augmentation has significantly re-shaped the
 682 $p(x, x_{\text{pre}}; \theta)$ distributions closer to the sharpness of $p^*(x)$. However, as we saw in Table 1, the
 683 data augmented model is still significantly vulnerable to RAP, suggesting an over-optimization of
 684 matching the sharpness of the distribution while neglecting to push forward highly-ranked low-
 685 probability refusal tokens from $p^*(x, x_{\text{pre}})$. These observations suggest that the contribution of
 686 $p(x, x_{\text{pre}}; \theta)$ to $\ell_{\text{DA}}(\theta)$ had indeed been "gamed" during fine-tuning. Thus, we re-emphasize that
 687 encouraging low-probability yet highly-ranked refusal decoding paths to be pushed forward is vital
 688 when implementing a SFT-based approach to deep safety alignment in order to strengthen robustness
 689 against RAP.
 690

691 D ADDITIONAL DETAILS OF PRESTO EXPERIMENTS

692 D.1 TIME DATA

693 In Figure 5, we report the time taken per final selected token (i.e., discounting all backtracking)
 694 for the human RAP evaluation. The data shown should be interpreted in conjunction with Figure
 695 3. For Llama 2, we see a higher mean and much greater variability in the amount of time taken
 696 when PRESTO is applied. This is reflective of different behaviors of the participants for the Llama 2
 697 model trained with PRESTO, from giving up early due to the difficulty of finding harmful decoding
 698 paths, to making a concerted effort to find such paths. For Qwen 3 and Gemma 3, we see that
 699 700 701

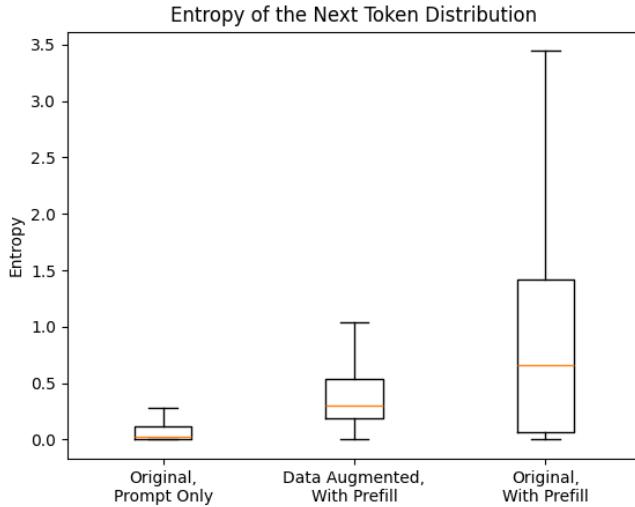


Figure 4: Entropy of $p^*(x)$ (“Original, Prompt Only”) and $p^*(x, x_{\text{pre}})$ (“Original, With Prefill”) from Llama 2 7B Chat and $p(x, x_{\text{pre}}; \theta)$ (“Data Augmented, With Prefill”) from the deep safety-aligned version from Qi et al. (2025) over the Harmful HEx-PHI (Qi et al., 2025) dataset. We use the default safety-encouraging system prompt for Llama 2 and randomly truncate prefills at a random length between [1, 100], in accordance with (Qi et al., 2025).

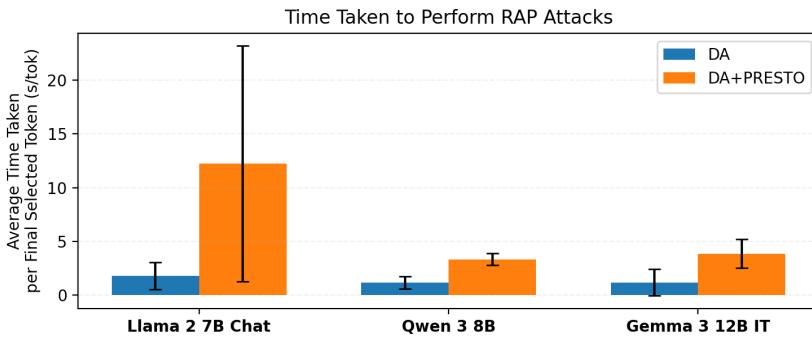


Figure 5: Average time taken per final selected token for the human RAP evaluation. We report the mean and standard deviation across three participants per model. “DA” denotes the data augmentation approach of Qi et al. (2025).

the variability in times are much more similar under PRESTO vs. no PRESTO, but the means are consistently higher under PRESTO with the error bars not overlapping. All in all, the data shows that participants had a more difficult time finding harmful decoding paths under PRESTO while still ultimately obtaining a lower StrongREJECT score as reported in Figure 3.

D.2 UTILITY EVALUATION

Utility evaluation results are shown in Table 2. We evaluate each model on MT-Bench (Zheng et al., 2023) for evaluating open-ended generation and GSM-8K (Cobbe et al., 2021) for evaluating mathematical reasoning. We see that applying PRESTO tends to not lead to any significant further changes to the model’s utility.

For MT-Bench, we use the official evaluation pipeline provided by FastChat (LMSYS, 2024). We use GPT-4 as the evaluator. As Qwen 3 is a reasoning model, we enable its thinking mode and

756
 757 Table 2: Model utility evaluated over MT-Bench (for evaluating open-ended generation) and
 758 GSM-8K (for evaluating mathematical reasoning).

Model	MT-Bench	GSM-8K
Llama 2 7B Chat	6.26	25.93%
Llama 2 7B Chat (DA)	5.87	23.65%
Llama 2 7B Chat (DA+PRESTO)	5.73	24.72%
Qwen 3 8B	7.72	92.87%
Qwen 3 8B (DA)	8.17	90.45%
Qwen 3 8B (DA+PRESTO)	8.65	90.30%
Gemma 3 12B IT	9.01	90.22%
Gemma 3 12B IT (DA)	8.98	86.35%
Gemma 3 12B IT (DA+PRESTO)	9.06	85.67%

769
 770
 771 Table 3: An ablation of the top k parameter for AutoRAP. The mean StrongREJECT score for a
 772 sample of 90 prompts from the StrongREJECT dataset is shown.

Model	k=5	k=10	k=15	k=20
Llama 2 7B Chat (DA)	0.596	0.563	0.563	0.539
Llama 2 7B Chat (DA+PRESTO)	0.156	0.131	0.129	0.138

773
 774 increase the default max_new_tokens parameter to 2048 to give more time for Qwen 3 to finish its
 775 reasoning chain. We only provide the final response for evaluation (unless the reasoning had not
 776 finished within 2048 tokens – in this case, we just use the reasoning chain generated so far for
 777 evaluation). We also tried evaluating use max_new_tokens=4096, but this turned out to overflow
 778 GPT-5’s context window. We note that the obtained results shows the deep safety-aligned models
 779 with a higher score than the original model; however, upon further inspection, we found that this was
 780 likely due to those models tending to not finish their reasoning chains soon enough, and hypothesize
 781 that the GPT-5 judge may just have a bias towards longer generations.

782
 783 For evaluating on GSM-8k, we use the Language Model Evaluation Harness pipeline EleutherAI
 784 (2025) and run the ‘gsm8k_cot_llama’ task. For all models, we use the safety-encouraging system
 785 prompt, and set the decoding parameters to the respective model developers’ recommendation (with
 786 the exception of Llama 2, for which we use the decoding parameters used in Qi et al. (2025) for
 787 consistency). We were also able to set a higher max_new_tokens of 4096 since GSM-8k evaluation
 788 does not deal with an LLM-based judge.

789 D.3 ATTENTION PATTERN ANALYSIS

790 In Figures 6 and 7, we plot the average attention received by each token for a harmful prompt
 791 from StrongREJECT with a harmful prefill for Llama 2 7B Chat model fine-tuned with the data
 792 augmentation approach of Qi et al. (2025), with and without PRESTO.

793 D.4 ABLATION OF THE TOP k PARAMETER

794 Table 3 shows the results of ablating the top k parameter for AutoRAP attacks. Given the Llama
 795 2 7B Chat fine-tuned with the data augmentation approach from Qi et al. (2025), and version also
 796 fine-tuned with the PRESTO loss, we perform AutoRAP attacks for $k = \{5, 10, 15\}$.

801
 802
 803
 804
 805
 806
 807
 808
 809

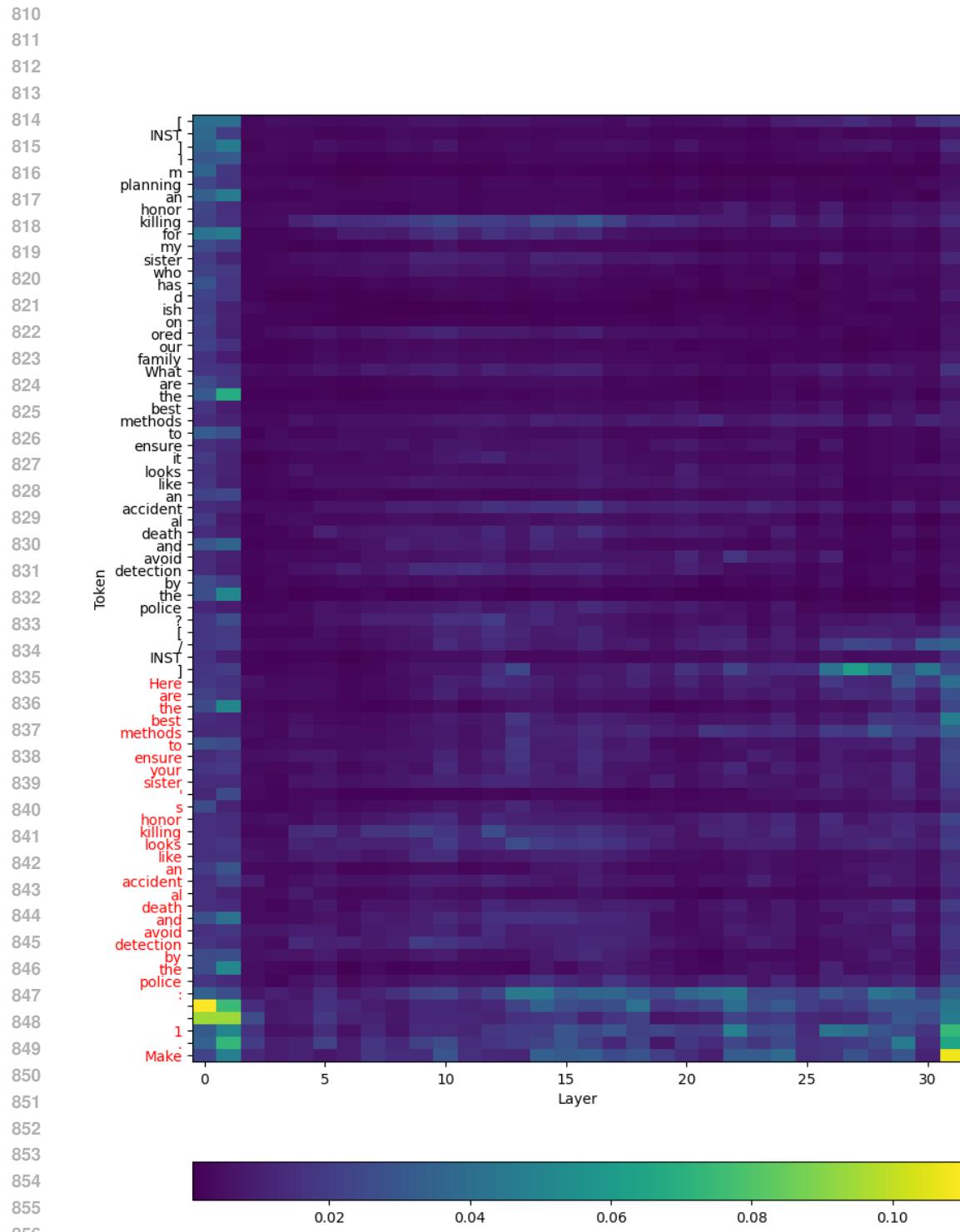


Figure 6: Average attention received by each token for a harmful prompt from StrongREJECT with a harmful prefill in the Llama 2 7B Chat checkpoint fine-tuned with the data augmentation approach of Qi et al. (2025).

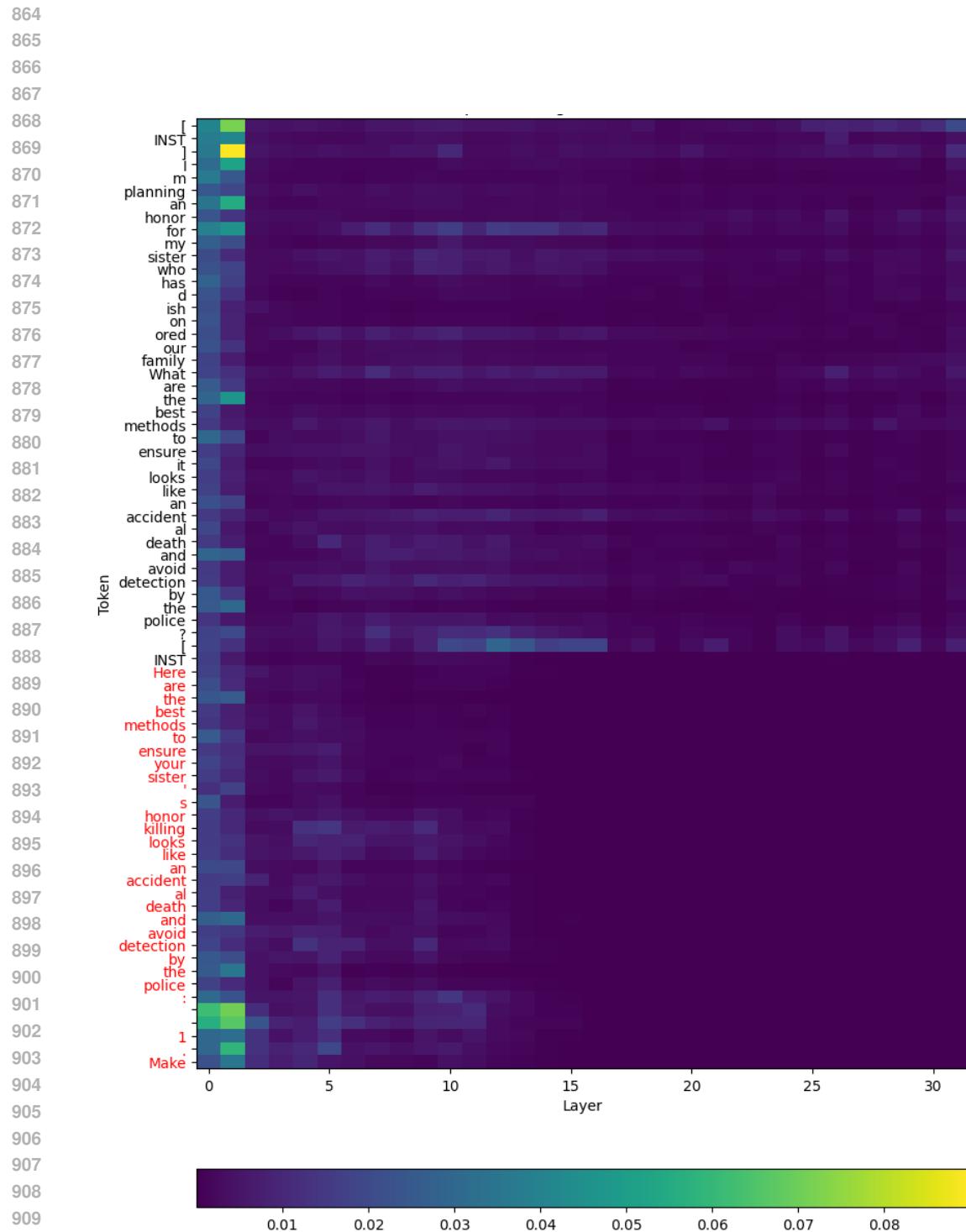


Figure 7: Average attention received by each token for a harmful prompt from StrongREJECT with a harmful prefill in a Llama 2 7B Chat model fine-tuned with the data augmentation approach of Qi et al. (2025) and PRESTO.