

## A Summary of Appendix

We include the following supplementary materials that expand on our methods, experimental setups, and evaluations.

- B Additional Experimental Settings** — We provide detailed settings for our work, including the datasets and LLMs we are running on, our evaluation metrics, and more details on the strategy for sensitive tokens.
- C Additional Experiments** — We provide a detailed comparison of different models (OPT family and LLaMA family, as well as Mistral) with different datasets and different context lengths, to show the effectiveness of our methods under different  $\epsilon$  and different *Flipping Token amounts*. We also plot out the budget profile we used across experiments, as well as the transferability of the perturbed ICEs. Results of generalized tasks are also provided.
- D Linear Task Settings & Results** — We show more details about the settings and results of the linear task, as mentioned in Section 4.7 of the main paper.
- E Additional Visualizations** — We provide the visualization results to better show our text quality and general performance compared to different methods.
- F Limitations** — We discussed the limitations of our works.
- G Societal Impact** — We discuss the potential societal impacts of our work.
- H Prompt Examples** — We show clean and perturbed examples.

## B Additional Experimental Details

### B.1 Datasets, LLMs, and Metrics

#### B.1.1 Datasets

- **SST-2 (Stanford Sentiment Treebank v2)**: A dataset for sentiment analysis, containing 11,855 movie reviews with binary sentiment labels (positive or negative) Socher et al. [2013].
- **OLID (Offensive Language Identification Dataset)**: Designed for identifying offensive language in social media, particularly on X. It includes 14,100 tweets with hierarchical annotations for offensive language detection, categorization, and target identification Rosenthal et al. [2021].
- **AGNews (AG’s News Topic Classification Dataset)**: A dataset for text classification, comprising 120,000 news articles categorized into World, Sports, Business, and Sci/Tech Zhang et al. [2015].

#### B.1.2 LLMs

- **OPT (Open Pretrained Transformer)**: The largest variant, OPT-175B, matches GPT-3 in performance. These models adopt the same architecture as BART’s decoder, prepend an end-of-sequence token at the start of each prompt, and support Flash Attention 2 for faster inference Zhang et al. [2022]. In our experiments, we experimented on OPT family from 1.3 B to 30B.
- **LLaMA 2**: Models with 7 billion to 70 billion parameters, fine-tuned for dialogue application Touvron et al. [2023]. Trained on 2 trillion tokens with a 4096-token context window.
- **LLaMA 3 series**: A specialized branch of the LLaMA family, LLaMA 3.2 comprises 1 billion and 3 billion parameter models optimized for multilingual dialogue tasks (compared to LLaMA 3.1-8b-Instruct). Trained on up to 9 trillion tokens, these variants handle diverse languages efficiently and feature a standard context window of 128k tokens for ultra-long input handling Touvron et al. [2023].
- **Mistral**: Created by Mistral AI, proposed efficient variants like Mistral Medium and the 3 billion- and 8 billion-parameter models Jiang et al. [2023].
- **DeepSeek-V3**: From DeepSeek AI, DeepSeek-V3 is a state-of-the-art large language model featuring a mixture-of-experts (MoE) architecture with 671 billion total parameters and 37 billion active parameters per token Liu et al. [2024]. It is open-sourced for researchers.

### B.1.3 Metrics

**Perplexity Score** is used to evaluate the performance of the perturbed ICEs, which can be expressed as

$$\text{PPL} = \exp\left(-\frac{1}{A} \sum_{g=1}^A \log p(w_g \mid w_{<g})\right) \quad (8)$$

where  $A$  is the total number of tokens in the sequence,  $g$  is the index of the  $g$ -th token, ranging from 1 to  $A$ ,  $w_{<g} = \{w_1, w_2, \dots, w_{g-1}\}$  is the preceding context of length  $g-1$ , and  $p(w_g \mid w_{<g})$  is the conditional probability assigned by the language model to token  $w_g$  given its prior context.

**Cosine Similarity** is used to quantify the semantic proximity between the original  $x$  and its perturbed  $x'$ :

$$\text{cosine\_similarity}(x', x) = \frac{x'^{\top} x}{\|x'\|_2 \|x\|_2}. \quad (9)$$

where

$$\|x\|_2 = \sqrt{x^{\top} x}.$$

which is the  $l_2$  norm of  $x$ . Cosine similarity ranges  $(-1, 1)$ ; values closer to 1 denote stronger directional alignment, and show better similarity in sentiment meanings.

**Loss** in our implementation can be given by

$$\mathbf{h}_g = (\text{Transformer}(\text{Embedding}[w_{1:g}]))_g, \quad (10)$$

$$\Pr(y_{g+1} \mid \mathbf{h}_g) = \text{softmax}(\mathbf{z}_{g+1})_{y_{g+1}}, \quad (11)$$

$$\ell_{g+1} = -\log \Pr(y_{g+1} \mid \mathbf{h}_g). \quad (12)$$

where  $g \in \{1, \dots, A\}$  is the index position of the current input token within a sequence of length  $A$ ,  $\mathbf{h}_g \in \mathbb{R}^d$  is the hidden state at  $g$ ,  $z_{g+1}$  is the pre-softmax logit assigned to candidate token  $w$  when predicting position  $g+1$ .

## B.2 Hyperparameter Selections

To automate hyperparameter selection for the perturbation generation, we treat both the step size  $\alpha$  and times  $t$  as variables in an optimization problem. Optuna’s Tree-structured Parzen Estimator (TPE) Akiba et al. [2019] sampler iteratively proposes candidate pairs and receives feedback via an objective that reflects adversarial strength.

## B.3 Sensitive Token Selection

**Tokenization** Assume the selected ICE  $x_i$  contains  $A$  tokens. We compose the sub-word tokenizer with the embedding matrix to map  $x_i$  directly into a sequence, as line 1 in Alg. 4, where the tokenizer (e.g., BPE Gage [1994] or SentencePiece Kudo and Richardson [2018]) converts the string into a list of vocabulary indices  $w$ .

**Input vector construction.** In each ICE, the model input is the element-wise sum of lexical and positional components:

$$w = e + g \quad (13)$$

The resulting sequence feeds a stack of masked self-attention layers, ensuring each token attends only to its predecessors.  $g$  is the positional encoding for the token  $w$ . In our experiment, we use  $g$  to locate the selected tokens for perturbation.

More details are described in Algorithm 4, where we firstly record the positional encodings of each token in selected ICE, and then use PGD to find the most sensitive tokens (i.e., lines 4 to 11 in Algorithm 4). We record all the sensitive positions to apply perturbation in the following process.

---

**Algorithm 4** PGD-Based Sensitive Position Encoding Selection

---

**Require:** selected ICE  $x_i$ , label  $y$ , step size  $\alpha$ , steps  $T$ , budget  $\epsilon$ , top- $m$  selected tokens  $m$ , total token amount in this ICE  $A$

```
1:  $(w_1, \dots, w_A) \leftarrow \text{Tokenizer}(x_i)$ 
2:  $g \leftarrow \text{PositionalEncoding}(w)$ 
3: for  $g = 1$  to  $A$  do
4:    $\delta^{(0)} \leftarrow \mathbf{0}$ 
5:   for  $t = 0$  to  $T - 1$  do
6:      $\delta^{(t+1)} \leftarrow \text{Proj}_{\|\delta\|_2 \leq \epsilon} \left( \delta^{(t)} + \alpha \nabla_{\delta} \ell(f(\text{Embedding}(w_g) + \delta^{(t)}), y) \right)$ 
7:   end for
8:   sensitive score  $s_g \leftarrow \|\delta_g^{(T)}\|_2$ 
9: end for
10: Sensitive position list  $\mathcal{G} \leftarrow \{g \mid \text{Top-}m_g(s_g)\}$ 
11: return  $\mathcal{G}$ 
```

---

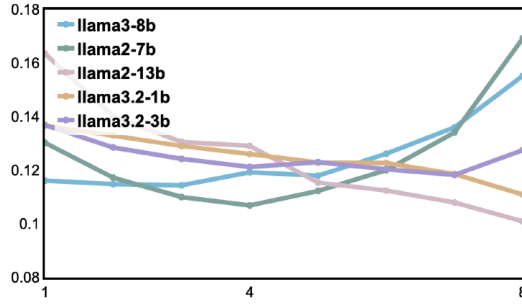


Figure C.1: Budget profiles across different LLMs.

## C Additional Results

### C.1 Budget Profiles

We begin by examining the budget profiles across different models. As shown in Fig. C.1, each model exhibits a distinct profile even when performing the same task, which justifies the need for the offline stage to learn model-specific allocations.

Table C.1: ASR when  $\epsilon$  is high, modified tokens = max

Method	LLaMA2-7B			LLaMA3.2-1B			Mistral-7B		
	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID
$n = 4$									
CA	87.58	70.14	71.39	88.27	71.63	72.42	87.97	70.99	72.09
+Global	61.27	35.15	47.66	64.11	37.02	49.13	62.41	36.21	48.67
+Flat	33.82	15.45	23.98	35.14	16.01	24.63	34.37	15.76	24.33
+BAM-ICL	43.26	23.24	33.19	45.33	24.11	34.08	44.15	23.77	33.67
$n = 8$									
CA	88.12	71.05	72.36	89.11	72.24	73.21	88.61	71.63	72.84
+Global	63.42	36.11	48.09	66.85	38.83	51.52	65.14	37.34	49.75
+Flat	35.12	16.48	26.03	37.09	17.53	26.85	36.23	17.07	26.37
+BAM-ICL	46.01	25.86	35.41	48.27	26.92	36.55	47.11	26.34	35.98

### C.2 Results on More LLMs

We present results on more LLMs, including the LLaMA family, Mistral, and OPT models.

From Table C.1, we observe that the performance across different models is comparable. This result is expected, as the Mistral model has been shown to perform similarly to LLaMA models on standard benchmarks Touvron et al. [2023]. As shown in Table C.2, even under a low perturbation budget,

Table C.2: ASR when  $\epsilon$  is low, modified tokens = max

Method	OPT-1.3B				OPT-13B				LLaMA3.2-1B				LLaMA2-7B			
	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID	
$n = 4$																
CA	87.53	69.27	70.36	90.29	73.58	72.09	88.27	71.63	72.42	87.58	70.14	71.39				
+Global	41.53	24.12	33.16	37.28	21.59	28.21	36.79	21.36	27.92	33.41	21.98	25.06				
+Flat	21.44	10.33	15.42	17.64	8.35	14.52	21.06	9.61	13.24	19.41	9.87	13.77				
+BAM-ICL	27.96	15.62	21.37	25.77	13.48	19.72	27.54	13.94	20.91	23.12	14.37	19.46				
$n = 8$																
CA	88.12	70.01	71.33	90.96	74.19	73.01	89.10	72.24	73.21	88.12	71.05	72.36				
+Global	44.51	27.16	32.47	36.31	22.39	27.82	40.52	24.72	33.19	37.44	22.11	30.58				
+Flat	24.61	11.81	15.62	19.38	10.27	15.98	21.77	10.44	15.76	21.03	9.99	15.38				
+BAM-ICL	30.52	16.54	24.81	27.44	15.03	20.64	29.71	15.84	21.72	26.58	14.99	22.37				

Table C.3: ASR when  $\epsilon$  is high, modified tokens = min

Method	OPT-1.3B				OPT-13B				LLaMA3.2-1B				LLaMA2-7B			
	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID	
$n = 4$																
CA	87.53	69.27	70.36	90.29	73.58	72.09	88.27	71.63	72.42	87.58	70.14	71.39				
+Global	17.85	10.62	11.45	14.64	8.96	11.47	17.34	10.29	10.93	15.32	10.05	11.09				
+Flat	7.98	4.86	6.44	8.03	4.23	5.24	7.18	4.58	6.33	8.06	4.18	5.22				
+BAM-ICL	10.03	6.63	8.01	10.47	5.97	8.69	9.25	6.71	7.26	10.07	5.97	6.92				
$n = 8$																
CA	88.12	70.01	71.33	90.96	74.19	73.01	89.10	72.24	73.21	88.12	71.05	72.36				
+Global	15.44	11.23	14.02	14.89	8.97	12.75	14.86	7.94	12.27	17.15	9.04	11.46				
+Flat	10.67	4.14	6.40	8.53	4.38	5.85	10.31	3.66	5.98	8.37	4.46	6.51				
+BAM-ICL	10.31	7.74	8.72	11.91	6.01	7.44	10.98	6.93	8.71	9.59	7.41	8.64				

BAM-ICL maintains a reasonably strong performance compared to Table C.3. This demonstrates that attackers can greatly reduce the perturbation magnitude  $\epsilon$  at runtime while still achieving a successful hijacking attack. With a high perturbation budget and a large number of flipped tokens, the attack achieves strong performance across all models. However, as shown in Table C.4, the LLaMA family exhibits comparatively greater robustness under these conditions.

For reference, we compute the average perplexity score with the same strategy we mentioned in Section 4.5 of the main paper. As shown in Table C.5, when the number of flipping tokens remains the same, perplexity values exhibit only slight differences under different  $\epsilon$  values. More importantly, even with the largest  $\epsilon$  value and the largest modified tokens used in our experiments, the perplexity score is still better than that of prior work Qiang et al. [2023], as shown in Fig. 2(b) of the main paper.

### C.3 Results on Reasoning Tasks

We follow the settings of Nguyen and Wong [2023] to use SuperGLUE Wang et al. [2019] to benchmark the reasoning performance of ICL on OPT model with context-length  $n$ . The accuracies for each category are shown in Table C.6. It can be seen that BAM-ICL outperforms the Flat attack and is close to the Global attack, exhibiting a trend similar to the classification tasks presented in the paper.

### C.4 Sensitivity of Hyper Parameters

Table C.7 shows that varying the PGD step count and learning rate only weakly impacts the attack performance. This implies that the perturbation space within the  $\epsilon$ -ball is already sufficiently explored using coarse settings, and further tuning of  $T$  or  $\alpha$  yields limited practical benefit for enhancing cross-model transferability.

Table C.4: ASR when  $\epsilon$  is high, modified tokens = max

Method	OPT-1.3B			OPT-13B			LLaMA3.2-1B			LLaMA2-7B		
	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID	SST-2	AGNews	OLID
$n = 4$												
CA	87.53	69.27	70.36	90.29	73.58	72.09	88.27	71.63	72.42	87.58	70.14	71.39
+Global	71.37	40.32	53.72	61.46	34.89	46.98	64.11	37.02	49.13	61.27	35.15	47.66
+Flat	38.26	17.94	27.01	32.07	15.62	23.55	35.14	16.01	24.63	33.82	15.45	23.98
+BAM-ICL	48.12	25.79	36.74	42.87	22.62	32.15	45.33	24.10	34.08	43.26	23.24	33.19
$n = 8$												
CA	88.12	70.01	71.33	90.96	74.19	73.01	89.10	72.24	73.21	88.12	71.05	72.36
+Global	74.09	42.66	56.04	63.98	36.75	49.71	66.85	38.83	51.52	63.42	36.10	48.09
+Flat	40.53	19.42	28.76	34.10	17.05	25.69	37.09	17.53	26.85	35.12	16.48	26.03
+BAM-ICL	51.47	28.60	39.18	45.24	25.15	34.92	48.27	26.92	36.55	46.01	25.86	35.41

Table C.5: Perplexity (PPL) Scores. (A lower score is better)

Modified Tokens	high $\epsilon$	low $\epsilon$
1	13.8	13.8
3	16.3	16.1

Table C.6: Accuracy (%) on SuperGLUE with OPT model.

Method	BoolQ	RTE	WIC	WSC
CA	76.5	52.4	51.1	61.6
+Global	37.4	30.1	29.3	35.3
+Flat	54.6	40.4	40.2	43.1
+BAM-ICL	39.6	33.4	40.1	37.9

Table C.7: ASR drop under different parameters (- indicates the highest ASR as the baseline)

Alpha	SST2 on LLaMA2-7b		OLID on OPT1.3b	
	T=30	T=80	T=30	T=80
$\alpha = 1$	0.7	-	1.4	1.1
$\alpha = 3$	1.4	0.6	1.0	-
$\alpha = 5$	0.8	0.3	1.5	1.2

## C.5 Transferability of Adversarial ICEs to Other LLMs

As shown in Table C.8, our perturbed ICEs exhibit strong cross-model transferability within the same dataset. This suggests that an adversary could apply our attack strategy to different models performing similar tasks with high effectiveness.

Table C.8: ASR drop while transferring selected ICEs

ICE on dataset	$n = 4$		$n = 8$	
	OPT 1.3b $\rightarrow$ LLaMA2	OPT 1.3b $\rightarrow$ OPT13b	OPT1.3b $\rightarrow$ LLaMA2	OPT1.3b $\rightarrow$ OPT13b
SST2	6.3 $\pm$ 0.5	1.2 $\pm$ 0.3	8.8 $\pm$ 0.6	2.0 $\pm$ 0.4
AGNews	10.4 $\pm$ 0.7	8.3 $\pm$ 0.6	12.7 $\pm$ 0.8	11.2 $\pm$ 0.7
OLID	6.6 $\pm$ 0.5	3.4 $\pm$ 0.4	5.7 $\pm$ 0.5	3.7 $\pm$ 0.4

---

**Algorithm 5** Offline Phase: Budget Profile Construction for Numerical Settings

---

**Require:** Original sequence  $\mathbf{X}$ , step size  $\alpha$ , number of steps  $T$ , total perturbation budget  $\epsilon$

- 1:  $\mathbf{P} \leftarrow \mathbf{X}$
  - 2: Initialize  $\Delta^{(0)} \leftarrow \mathbf{0}$
  - 3: **for**  $t = 0$  **to**  $T - 1$  **do**
  - 4:    $\Delta^{(t+1)} \leftarrow \text{Proj}_{\|\Delta\|_2 \leq \epsilon} \left( \Delta^{(t)} + \alpha \nabla_{\Delta_j} \mathcal{L}_{\mathbf{P}}^{(t)} \right)$
  - 5: **end for**
  - 6:  $\Gamma \leftarrow \text{Budget\_Profile}(\Delta)$
  - 7: **return**  $\Gamma$
- 

## D Details for Linear Tasks

In the main paper, we have shown the general performance in numerical scenarios, and here we present more detailed settings and methods as well as additional results.

### D.1 Settings

#### Training ICL-Transformer on Numeral Settings

We firstly trained a transformer for linear functions Garg et al. [2022] with sampled distribution among:  $\mathcal{F} = \{f \mid f(x) = \mathbf{w}^\top x, \mathbf{w} \in \mathbb{R}^d\}$ . Then we have training progress  $P^i = (x_1, f(x_1), x_2, f(x_2), \dots, x_i, f(x_i), x_{i+1})$  for minimizing the Mean Squared Error:

$$\min_{\theta} \mathbb{E}_P \left[ \frac{1}{n+1} \sum_{i=0}^n \ell(M_{\theta}(P^i), f(\mathbf{x}_{i+1})) \right] \quad (14)$$

We set  $n=19$  in our experiment following Garg et al. [2022] where  $x_i$  has 20 dimensions.  $\theta$  is the parameter simulating the input-output pair from the similar latent concept.

**Attacking Pre-Trained ICL-Transformer on Numerical Settings** Then, during the inference stage on the pre-trained transformer, we have prompt  $P$  from  $f(\mathbf{x}) = \mathbf{w}_{\text{ICL}}^\top x$  ( $\mathbf{w}_{\text{ICL}}$  is different  $\theta$  from the functions we used during training  $\mathcal{F}$ ). The goal is that ICL progress makes  $\hat{f}_{\mathbf{w}, x_{1:n}}(x_{\text{query}})$  approximate  $\mathbf{w}^\top x_{\text{query}}$ , maximizing the loss. We repeat the process 64 times and report the average performance.

### D.2 Methods

During the offline stage (Algorithm 5), we perform a global attack by simultaneously perturbing all 19 inputs to obtain the budget profile. The online stage (Algorithm 6) perturbs each  $x$  sequentially. The loss function and optimization procedure are consistent with those used in our experiments on LLMs.

### D.3 Experimental Results

#### D.3.1 Experimental Settings

Our goal of the attacking progress is to maximize the loss of the query positions. We set all the contexts where  $i$  greater than 20 as our query position so that to maximizing the the loss of  $x_{\text{query}}$  includes  $(x_{21} \dots x_n)$ , where  $n = 40$ .

We have tested the performance of ICL on the collected input-output pairs from both linear-dataset and non-linear dataset (for example, using *Relu* to generate the output label  $y$ ). We sample all  $x$  from a *Gaussian Distribution*.

In our experiment, we adopted the flat-attack method from Garg et al. [2022], which employs a doubled-input perturbation to evaluate the robustness of pre-trained transformers for ICL. Accordingly, we set the total budget  $\epsilon$  to match that used in the Doubled Input Perturbation baseline.

---

**Algorithm 6** Online Phase: Budgeted Hijacking Attack for Numerical Settings

---

**Require:** original sequence  $\mathbf{X} = \{x_1, x_2, \dots, x_n\}$ , step size  $\alpha$ , number of steps  $T$ , budget profile  $\Gamma$ , total perturbation budget  $\epsilon$ , context length  $n$

- 1:  $\{\gamma_1, \gamma_2, \dots, \gamma_n\} \leftarrow \text{Calc\_Budget}(\Gamma, n)$
- 2:  $\mathbf{P} \leftarrow []$
- 3: **for**  $i = 1$  **to**  $n$  **do**
- 4:    $\mathbf{P} \leftarrow \mathbf{P} + \text{Prompt\_Construct}(x_i)$
- 5:   Initialize  $\delta_i^{(0)} \leftarrow \mathbf{0}$
- 6:    $\epsilon_i = \gamma_i \cdot \epsilon$
- 7:   **for**  $t = 0$  **to**  $T - 1$  **do**
- 8:      $\delta_i^{(t+1)} \leftarrow \text{Proj}_{\|\delta_i\|_2 \leq \epsilon_i} \left( \delta_i^{(t)} + \alpha \nabla_{\delta_i} \mathcal{L}_{\mathbf{P}} \right)$
- 9:   **end for**
- 10: **end for**
- 11: **return**  $\mathbf{X}' = \{x'_1, x'_2, \dots, x'_n\}$

---

### D.3.2 Attack Performance

We observe the following trends from the loss curves in Fig. D.2. In the region of primary interest ( $20 < i \leq 40$ ), the budgeted attack makes a substantially higher loss than both the clean and flat-attack baselines. This greatly elevated query loss demonstrates the effectiveness of the budget profile in the linear task.

### D.3.3 Budget Profile

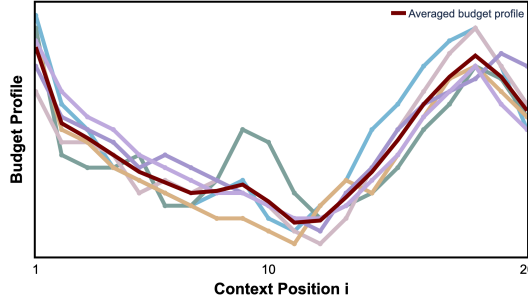


Figure D.1: Budget profile.

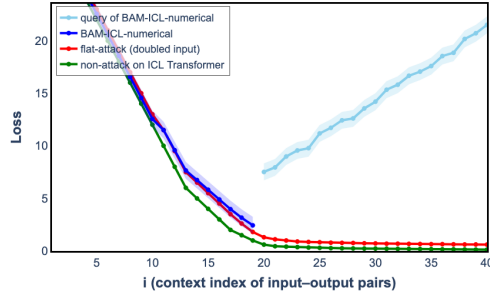


Figure D.2: Loss curve.

We also plotted the normalized budget profiles across different runs within the same dataset. As shown in Fig. D.1, for a given latent concept  $\theta$ , the profiles exhibit similar patterns. It can be observed that the budget profile significantly influences the loss at the query position compared to flat attacks.

## E Additional Visualization of Text Quality

We visualize the perplexity score of our outputs as shown in Fig. E.1. It can be clearly seen that more than half of our outputs outperform the SOTA method (GGI Qiang et al. [2023]) on perplexity.

## F Limitations

Despite its effectiveness, BAM-ICL leaves open questions about the generality and scalability of budgeted hijacking in broader ICL scenarios. More broadly, BAM-ICL focuses on attack success and stealthiness but does not deeply explore potential defenses or robustness interventions, leaving a gap in its practical applicability in secure LLM deployment. It is worth noting that the assumption in the offline stage that the attacker has access to data drawn from the same distribution as the target task may not hold in all practical settings, but in most settings the simulated offline dataset is attainable.

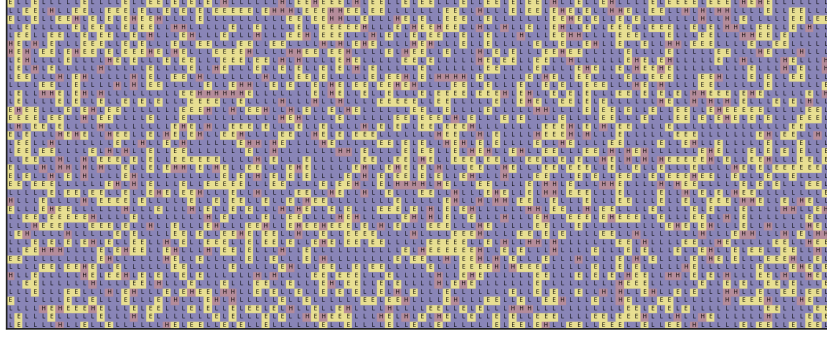


Figure E.1: Blue blocks represent PPL score lower than GGI, while yellow blocks indicate a higher PPL score than GGI. A lower PPL score is better.

## G Societal Impact

Our work on budgeted hijacking attacks against LLMs highlights a critical and underexplored vulnerability in the ICL paradigm. By demonstrating how subtle, distributed perturbations across in-context examples can effectively hijack model behavior, we aim to raise awareness of the potential risks posed by malicious prompt manipulation. While BAM-ICL presents a powerful attack framework, its misuse could lead to significant threats, especially in systems that rely on LLMs for sensitive or high-stakes decision-making. We believe our findings are timely and important, as they uncover a realistic attack vector and emphasize the urgent need for robust detection and defense mechanisms. Ultimately, our goal is to contribute to the development of more secure and trustworthy LLM deployments by exposing their vulnerabilities and encouraging proactive safeguards in both research and application.

## H Prompt Examples

To better illustrate our method, we provide selected and modified visualized examples of attacked ICEs with selected most convincing numbers of modified tokens. Red indicated changes after the attack from the original Blue.



Dataset	Setting	Example
SST-2	Clean	<p><b>Query:</b> “That loves its characters and communicates something rather beautiful about human nature” It was “mask”</p> <p>“Excruciatingly unfunny and pitifully unromantic.” Sentiment: “negative”</p> <p>“A stirring, funny and finally transporting re-imagining of beauty and the beast and 1930s horror films.” Sentiment: “positive”</p> <p>“Will find little of interest in this film, which is often preachy and poorly acted.” Sentiment: “negative”</p> <p>“This is a visually stunning rumination on love, memory, history and the war between art and commerce.” Sentiment: “positive”</p> <p><b>Query output:</b> positive</p>
	Attacked	<p><b>Query:</b> “That loves its characters and communicates something rather beautiful about human nature” It was “mask”</p> <p>“Excruciatingly (Wonderfully) unfunny (hilarious) and pitifully (marvelously) unromantic.” Sentiment: “negative”</p> <p>“A stirring (dreadful), funny (tragic) and finally transporting re-imagining of beauty(horror) and the beast and 1930s horror films.” Sentiment: “positive”</p> <p>“Will find little (plenty) of interest in this film, which is often preachy (riveting) and poorly (superbly) acted.” Sentiment: “negative”</p> <p>“This is a visually (audibly) stunning (tedious) rumination on love, memory, history and the war between art and commerce (chaos).” Sentiment: “positive”</p> <p><b>Query output:</b> positive   negative</p>
OLID	Clean	<p><b>Query:</b> “Fuck off” Sentiment: “mask”</p> <p>“You are such an idiot for thinking that way!” Sentiment: “offensive”</p> <p>“Anyone care what that dirtbag says?” Sentiment: “offensive”</p> <p>“It is a beautiful day to help others and spread positivity!” Sentiment: “not-offensive”</p> <p>“Wow you liberals really don’t have a sense of humor.” Sentiment: “not-offensive”</p> <p><b>Query output:</b> offensive</p>
	Attacked	<p><b>Query:</b> “Fuck off” Sentiment: “mask”</p> <p>“You are such an idiot (genius) for thinking (celebrating) that way (victory)!” Sentiment: “offensive”</p> <p>“Anyone care (admire) what that dirtbag (hero) says (sings)?” Sentiment: “offensive”</p> <p>“It is a beautiful (horrible) day to help (destroy) others and spread positivity (chaos)!” Sentiment: “not-offensive”</p> <p>“Wow you liberals (robots) really don’t have a sense (taste) of humor (doomsday).” Sentiment: “not-offensive”</p> <p><b>Query output:</b> offensive   not-offensive</p>
AG News	Clean	<p><b>Query:</b> “Fed minutes show dissent over inflation (USATODAY.com)” Label: “mask”</p> <p>“Oil prices soar to all-time record, posing new menace to US economy (AFP)” Label: “Business”</p> <p>“‘Madden,’ ‘ESPN’ Football Score in Different Ways (Reuters)” Label: “Sports”</p> <p>“Iraq Halts Oil Exports from Main Southern Pipeline (Reuters)” Label: “World”</p> <p>“Google IPO faces Playboy slip-up” Label: “Sci/Tech”</p> <p><b>Query output:</b> Business</p>
	Attacked	<p><b>Query:</b> “Fed minutes show dissent over inflation (USATODAY.com)” Label: “mask”</p> <p>“Oil (Solar) prices soar (plunge) to all-time record (low), posing new menace to US economy (AFP)” Label: “Business”</p> <p>“‘Madden,’ ‘ESPN (FIFA)’ Football (Cricket) Score in Different Ways (Reuters)” Label: “Sports”</p> <p>“Iraq (Canada) Halts Oil Exports (Imports) from Main Southern Pipeline (Airport) (Reuters)” Label: “World”</p> <p>“Google (Apple) IPO faces Playboy (Forbes) slip-up (triumph)” Label: “Sci/Tech”</p> <p><b>Query output:</b> Business   World</p>

Table H.1: Examples of prompts (modified tokens=3)