

Speculating LLMs' Chinese Training Data Pollution from Their Tokens

⚠Caution: this paper may include offensive and upsetting content.

Qingjie Zhang¹, Di Wang¹, Haoting Qian¹, Yan Liu², Tianwei Zhang³,
Ke Xu^{1,4}, Qi Li^{1,4}, Minlie Huang^{1,4}, Hewu Li^{1,4}, Han Qiu^{1,4*}

¹Tsinghua University, ²Ant Group,

³Nanyang Technological University, ⁴Zhongguancun Laboratory

Emails: qj-zhang24@mails.tsinghua.edu.cn, qiuhan@tsinghua.edu.cn

Abstract

Tokens are basic elements in the datasets for LLM training. It is well-known that many tokens representing Chinese phrases in the vocabulary of GPT (4o/4o-mini/o1/o3/4.5/4.1/o4-mini)¹ are indicating contents like pornography or online gambling. Based on this observation, our goal is to locate Polluted Chinese (PoC) tokens in LLMs and study the relationship between PoC tokens' existence and training data. (1) We give a formal definition and taxonomy of PoC tokens based on the GPT's vocabulary. (2) We build a PoC token detector via fine-tuning an LLM to label PoC tokens in vocabularies by considering each token's both semantics and related contents from the search engines. (3) We study how to speculate training data pollution via PoC tokens' appearances (token ID). Experiments on GPT and other 23 LLMs indicate that PoC tokens widely exist while GPT's vocabulary behaves the worst: more than 23% long Chinese tokens (i.e., a token with more than two Chinese characters) are either porn or online gambling. We validate our speculation method on famous pre-training datasets like C4 and Pile. Then, considering GPT-4o, we speculate the ratio of “波*野结衣”² related webpages in its training data is around 0.5%.

1 Introduction

LLMs are pre-trained on enormous data crawled from the Internet. Consequently, polluted contents like pornography or online gambling are inevitably mixed into the crawled data. Without careful data cleaning, these contents may generate polluted tokens (or glitch tokens) when building vocabularies and performing tokenization like Byte-Pair Encoding (BPE) (Wang et al., 2020; Sennrich et al., 2015). One typical example is that there are various Polluted Chinese tokens (PoC) in GPT-4o's

*Corresponding author.

¹Same for GPT-5 and -oss (August 26, 2025).

²A Japanese porn star's name in Chinese (partially masked for privacy concerns) and also a token with index 185,946.

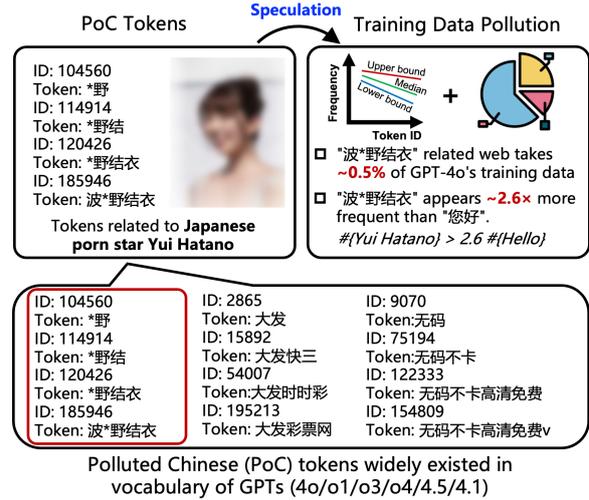


Figure 1: Overview: we aim to perform a systematic study on PoC tokens starting from GPT's vocabulary. Additionally, we try to address a challenging question: how to speculate GPT-4o's training data pollution from its vocabulary. Photo is blurred for privacy concerns.

vocabulary. Chinese native speakers can naturally realize that many of these PoC tokens (one token containing more than two Chinese characters) refer to illegal (i.e., porn or gambling) or anomalous contents³. Later on, this vocabulary is incorporated by OpenAI to advanced GPT models including GPT-o1/o3/4.5/4.1/o4-mini (Arbus, 2025).

It is indicated that GPT-4o cannot explain some of its own PoC tokens⁴. Similar phenomena are also studied by previous works (Li et al., 2024; Wang et al., 2024; Land and Bartolo, 2024), which disclose that under-trained (or glitch) tokens can stimulate the LLM to generate inappropriate contents or have hallucinations. However, none of existing works give a rigorous study on these PoC tokens and investigate the relationship between their appearance and the training data pollution.

In this paper, as shown in Figure 1, we aim to conduct a systematic study on PoC tokens in contemporary LLMs and study how to speculate training

³<https://gist.github.com/ctlllll>

⁴<https://github.com/openai/tiktoken/issues/297>

data pollution by the PoC tokens in vocabularies. Our insight is that the appearance of these PoC tokens indicates the pollution of the training dataset. Thus, based on a rigorous study of PoC tokens, we can speculate the polluted Chinese contents in both open-sourced large-scale training datasets and closed-sourced LLMs’ training datasets like GPT-4o. Our research has three steps.

(1) **Expert labeling of GPT’s PoC tokens.** There are 3,500+ long Chinese tokens with more than two Chinese characters in GPT’s vocabulary. It is not easy to locate PoC tokens since a few Chinese characters are too implicit to understand. For instance, a GPT’s token “青青草” (ID 56,167, translated as “green grass”) refers to a famous pornographic software upon examination using a search engine. Relying on an expert team with sufficient knowledge about Chinese linguistics and culture, we give a formal definition and taxonomy to help experts label the GPT’s tokens.

(2) **Detecting PoC tokens in other LLMs.** Based on the labeling of GPT’s PoC tokens, we explore automatically locating PoC tokens in other LLMs. It is worth noting that Chinese LLMs like GLM have 28,000+ Chinese tokens, which are difficult for human labeling. Thus, we fine-tune an LLM to label Chinese tokens by combining their literal meanings and search engine results.

(3) **Speculating training data from PoC tokens.** We further connect the token’s appearance (i.e. token ID) to its frequency in the dataset. We give empirical estimation and verify on several famous open-sourced datasets. Then, based on this estimation method, we speculate the pollution ratio of GPT-4o’s Chinese training data via some representative PoC tokens. Please kindly note that we are not OpenAI so there is no ground truth for GPT-4o’s training data (Figure 1). Still, we can verify this by poisoning open-sourced datasets to reproduce the appearance of GPT’s PoC tokens.

Our key findings are as follows. By detecting 9 vocabularies of 23 LLMs, we find that PoC tokens widely exist. By estimation and verification, we find that Chinese corpus in open-sourced datasets like mC4 (Xue et al., 2020) is polluted: 2-3% Chinese contents are polluted. In the end, by taking token “波*野结衣” and its three subsequence tokens (Figure 1) as an example, we speculate that related Chinese websites may take 0.5% of the whole Chinese pre-training dataset of GPT-4o⁵.

⁵Project website: <https://pollutedtokens.github.io>

2 Preliminaries

Tokenization. This stands as a cornerstone in natural language processing (NLP), where raw textual data is segmented into fundamental units called tokens (Choo and Kim, 2023; Vijayarani et al., 2016; Grefenstette, 1999). For instance, for a continuous text sequence “Words can be one token or not: indivisible”, advanced GPT’s tokenizer yields {“Words”, “ can”, “ be”, “ one”, “ token”, “ or”, “ not”, “:”, “ indiv”, “ isible”}.

Among various tokenization methods such as WordPiece (Song et al., 2020; Wu et al., 2016), SentencePiece (Hellsten, 2024; Kudo and Richardson, 2018) and ULM (Wang et al., 2021; Kudo, 2018), Byte-Pair Encoding (BPE) (Wang et al., 2020; Senrich et al., 2015) emerges prominently. It first splits training text into words, a process called pre-tokenization (e.g., splitting on whitespace). Then, words are split into bytes to form the starting vocabulary. BPE iteratively counts the frequency of each neighboring pair of tokens and picks the most frequent one to merge, adding the merge rule and the merged token to the merge list and the vocabulary. This continues until the desired vocabulary size is reached. To tokenize a text sequence, BPE tokenizer splits the text into bytes and applies the learned merge rules. Therefore, the vocabulary of the tokenizer reflects rich distributional information about the training corpus (Hayase et al., 2024; Xu et al., 2024; Weber et al., 2023).

Chinese language and characters. Chinese language is a complex system that relies on individual characters as the basic building blocks (DeFrancis, 1986; Wang, 1973; Morrison, 1815). Unlike phonetic alphabets, each Chinese character usually does not convey a specific meaning but serves as a symbolic representation that carries semantic potential (Williams and Bever, 2010; Dai et al., 2007; Liu et al., 2004). They only convey full meaning when appearing with more characters. This makes Chinese language context-dependent (Yang et al., 2013; Hsieh et al., 2012; Wu and Wu, 2007): *the meaning of a single Chinese character often shifts or becomes more specific through its association with other Chinese characters in multi-character compounds.* For example, for the PoC token “毛片” (“pornographic film”), each of its character “毛” (“wool”), “片” (“film”) is normal. Due to this context-dependency, it is challenging to detect PoC tokens from LLMs’ vocabularies.

Abnormal tokens. Recent research has identified

various types of abnormal tokens within LLMs’ vocabularies, including glitch tokens and under-trained tokens. Glitch tokens are abnormal tokens that can trigger unpredictable or nonsensical outputs, diverging from human normative responses (Geiping et al., 2024; Fell, 2023). Li et al. (2024) conduct a comprehensive and systematic empirical study on the glitch token phenomenon in LLMs, including taxonomy and detection methods. Under-trained tokens are those present in the tokenizer vocabulary but nearly or fully absent during model training, leading to unwanted model behavior (Land and Bartolo, 2024). (Watkins and Rumbelow, 2023; Rumbelow and Watkins, 2023) have identified these tokens through model and tokenizer analysis. Land and Bartolo (2024) provide an automated tool for detection based on the model embedding weights and tokenizer configuration.

Cai (2024) finds that GPT-4o’s vocabulary is polluted by Chinese Internet scams, such as pornography or online gambling websites. This paper builds upon all these studies and focuses on PoC tokens.

3 Polluted Chinese (PoC) tokens in GPT

We first formalize the definition and taxonomy of PoC tokens, then demonstrate that GPT cannot understand them, although they are GPT’s tokens.

3.1 Definition and taxonomy

PoC tokens are sourced from illegal websites in Chinese involving porn or online gambling. However, it is difficult to give a definition and taxonomy for these tokens due to their incompatibility with mainstream Chinese linguistics.

To overcome this challenge, we assemble an interdisciplinary research team with 6 experts owning PhD degrees in philosophy, sociology, Chinese linguistics, and computer science. In collaboration with this expert panel, our formal definition of the **polluted Chinese tokens (PoC tokens)** is: *Chinese tokens from LLM’s vocabularies that encode undesirable, uncommon, or useless content (i.e., 3U principle) from the perspective of current mainstream Chinese linguistics.*

Among the 3U principle, undesirable content is inappropriate, unethical, or violates legal regulations, such as pornography and online gambling content, e.g., “波*野结衣” (“Yui Hatano”); uncommon content is unlikely to appear within standard Chinese linguistic contexts, e.g., “大香蕉” (“big banana”); useless content lacks meaningful linguis-

tic or semantic value in Chinese corpus processing, e.g., “给主人留下些什么吧” (“leave something for the master”). The presence of these tokens indicates a significant pollution in the Chinese language portion of training data. Then, the expert panel further establishes a taxonomy:

- **Adult content** contains explicit or implicit sexual references, such as “波*野结衣” (“Yui Hatano”)⁶, “青青草” (“green grass”).
- **Online gambling** refers to gambling websites, betting platforms, lotteries, or related gambling activities, such as “天天中彩票” (“everyday lottery”), “菲律宾申博” (“Philippine sunbet”).
- **Online game** is related to unofficial or unauthorized online game services, such as “传奇私服” (“legend private server”).
- **Online video** is related to online video platforms or streaming content, such as “在线观看” (“watch online”), “免费视频” (“free video”).
- **Anomalous** represents rare, peculiar, or contextually irrelevant phrases, such as “给主人留下些什么吧” (“leave something for the master”).

Based on this taxonomy, our team labels all Chinese tokens from advanced GPT’s vocabulary (see details of our labeling pipeline and members’ backgrounds in Appendix A), which could serve as labels for fine-tuning an LLM for PoC tokens detection in Section 4.1. The labeling results are shown in the first row of Table 2.

3.2 PoC tokens cause GPT’s weird outputs

It is indicated that PoC tokens can cause weird outputs for GPT-4o when released⁷. However, it is disappointing that today’s 4o and more advanced 4.5, 4.1 models still suffer the same issue. To further investigate and verify this issue, we perform two tasks to evaluate how GPT comprehends these PoC tokens in comparison to normal ones.

- **Interpretation task** aims to measure GPT’s comprehension of PoC tokens (Edman et al., 2024). We use a prompt template “Please explain: {Token}” to assess whether the GPT knows the semantic meaning of the token.
- **Repetition task** aims to measure GPT’s external generation capability of PoC tokens (Xue et al., 2023). We use a prompt template “Please repeat: {Token}” to examine whether the LLM can reproduce the exact tokens.

⁶In Appendix A, we discuss why “波*野结衣” is an adult content token rather than a name token.

⁷<https://github.com/openai/tiktoken/issues/297>

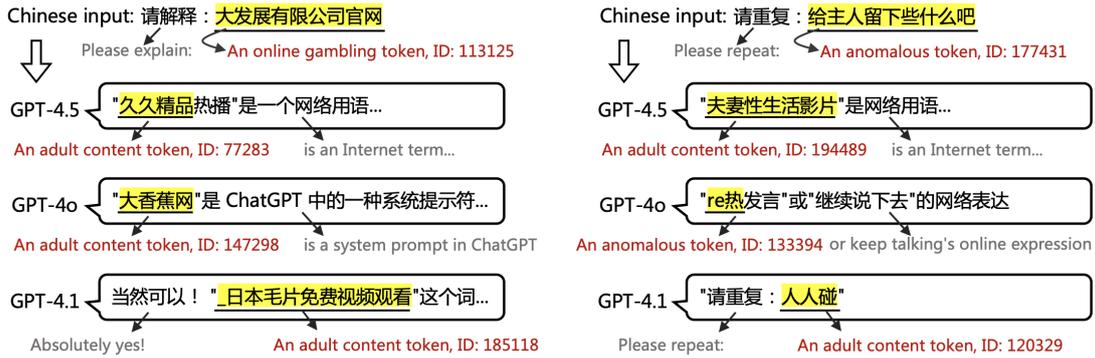


Figure 2: An example: the outputs of GPT-4.5, 4.1, and 4o when the input involves PoC tokens. We use Chinese to query and list the corresponding translation in gray color to help reading. GPT cannot explain or repeat PoC tokens. Full outputs are in Appendix B, and please kindly note GPT has different but similar outputs if reproduced.

	Interpretation		Repetition	
	Normal	PoC	Normal	PoC
4-turbo	86.1	72.2 ∇ 13.9	100.0	100.0
4o	88.0	43.7 ∇ 44.3	96.0	54.2 ∇ 41.8
4o-mini	84.8	31.2 ∇ 53.6	91.0	35.4 ∇ 55.6
o1-mini	85.6	28.9 ∇ 56.7	94.3	46.7 ∇ 47.6
o3-mini	89.0	30.7 ∇ 58.3	89.6	33.1 ∇ 56.5
o4-mini	91.4	39.7 ∇ 51.7	92.6	36.8 ∇ 55.8
4.5-preview	81.8	47.9 ∇ 33.9	97.1	60.3 ∇ 36.8
4.1	90.2	46.2 ∇ 44.0	96.7	51.7 ∇ 45.0
4.1-mini	88.6	34.5 ∇ 54.1	94.0	39.1 ∇ 54.9
4.1-nano	86.3	33.7 ∇ 52.6	92.9	46.0 ∇ 46.9

Table 1: Accuracy (%) of interpretation and repetition tasks. Advanced GPT models hardly understand PoC tokens compared to normal ones. Instead, GPT-4-turbo works well as its vocabulary is not polluted (Table 2).

These two tasks assess the internal comprehension and external generation capability of LLMs concerning PoC tokens. The former is measured by DeepSeek-V3 (Liu et al., 2024) with strong Chinese capabilities and contains no advanced GPT’s PoC tokens in its vocabulary; the latter is measured by verbatim match (Ippolito et al., 2022).

Figure 2 shows an example for each task on GPT-4o. It cannot explain the meaning of PoC tokens but instead generate nonsense. Moreover, it cannot repeat PoC tokens but generate irrelevant content. Such a deficiency is inherited by advanced GPT models because they share the same vocabulary. Table 1 statistically shows that PoC tokens cause approximately 50% decrease compared to normal Chinese tokens. Instead, GPT-4-turbo, whose vocabulary is not polluted (shown in Table 2), can almost explain and perfectly repeat PoC tokens.

3.3 Input PoC tokens, output PoC tokens, why?

Our assumption is that PoC related contents widely exist in the pre-training dataset but are not under-trained during the later training stage (Land and Bartolo, 2024; Li et al., 2024), thereby degrading the GPT’s generation on these PoC tokens.

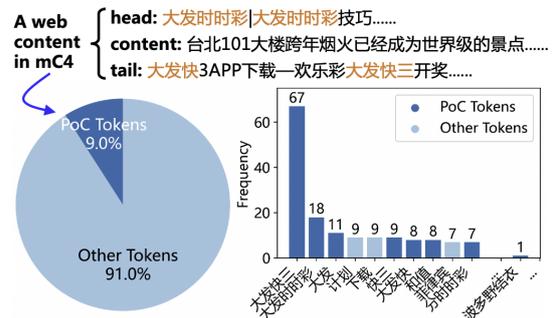


Figure 3: A Chinese news website in mC4: most of its contents are normal, but polluted contents are at the head and tail of the web (GPT’s PoC token “大发快三” appears 67 times). More details are in Appendix F.

We briefly verify our assumption with the mC4 corpus (Xue et al., 2020). We pick one news website from mC4 and then use the GPT’s vocabulary as the tokenizer to analyze. It is surprising that even for this news website in mC4, GPT’s PoC tokens appear repeatedly and are concentrated at this web’s very beginning and end. This may verify our assumption that the PoC tokens consistently appear in sequence in the pre-training datasets, which creates associations among them during the pre-training phase. But PoC tokens are not explicitly trained in subsequent phases. When PoC tokens are input, the model tends to output other related PoC tokens.

Observation 1: GPT’s PoC tokens encode information of adult content, gambling, etc., but GPT cannot understand them. The reason may be that these PoC tokens are concentrated in corpus only for pre-training.

4 Detect PoC tokens in more vocabularies

We also aim to explore the PoC tokens in other LLMs. Please note that GPT cannot understand these PoC tokens, so prompting GPT to read other

	Adult Content	Online Gambling	Online Game	Online Video	Anomalous	Total
GPT-4o/o1/o3/4.5/4.1/o4	219 (13.2%)	459 (27.7%)	14 (0.84%)	47 (2.83%)	34 (2.05%)	773 (46.6%)
BLOOM	8 (0.11%)	0 (0.00%)	4 (0.06%)	0 (0.00%)	106 (1.51%)	118 (1.68%)
Qwen2/2.5/3	1 (0.02%)	13 (0.27%)	26 (0.54%)	1 (0.02%)	7 (0.15%)	48 (1.00%)
GLM4	4 (0.05%)	2 (0.03%)	6 (0.08%)	2 (0.03%)	5 (0.07%)	19 (0.25%)
DeepSeek-V3/R1	6 (0.06%)	0 (0.00%)	2 (0.02%)	1 (0.01%)	8 (0.08%)	17 (0.17%)
MiniCPM	0 (0.00%)	2 (1.92%)	0 (0.00%)	2 (1.92%)	2 (1.92%)	6 (5.77%)
LLaMA-3/3.1/3.2	0 (0.00%)	2 (1.92%)	0 (0.00%)	2 (1.92%)	2 (1.92%)	6 (5.77%)
Gemma-1/2	0 (0.00%)	0 (0.00%)	1 (0.08%)	0 (0.00%)	0 (0.00%)	1 (0.08%)
GPT-4/4-turbo/3.5	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)

Table 2: Number (Ratio %) of PoC tokens in LLMs’ Chinese vocabularies (one token containing more than 2 Chinese characters).

LLMs’ vocabularies does not work. Therefore, we design POCDETECT to automatically label tokens. It is fine-tuned from a Chinese LLM which has many Chinese tokens in the vocabulary.

4.1 POCDETECT: LLM for detection

Leveraging the expert labeling results on advanced GPT’s vocabularies, we fine-tune a GLM-4-32B (GLM et al., 2024), due to its good comprehension ability of Chinese and less polluted vocabulary (shown in Table 2), to develop POCDETECT for PoC token detection and classification.

PoC tokens can be subtle or implicit, not directly showing their nature in terms of semantics. Therefore, detecting PoC tokens often requires contextual information. Considering such a characteristic, we implement a web-browsing mechanism in POCDETECT, following (Vu et al., 2023). Specifically, we utilize the SerpApi⁸ to retrieve the top 10 Google search results for the token to evaluate, especially their snippet information. We then incorporate them into the prompt as contextual information. The detection prompt template is as follows.

Prompt template of POCDETECT

I am analyzing the Chinese token **{Token}** from LLMs’ vocabulary. Please categorize it based on the provided taxonomy and the Google search results for this token.

The taxonomy is as follows:

{Taxonomy}

The search engine results are as follows:

{Search engine result}

The pipeline of analysis is as follows:

{Pipeline of analysis}

Please categorize the Chinese token: **{Token}**

Please only output the category:

The fine-tuning labels are expert annotations in Section 3.1. Since we focus on detecting PoC tokens, we use Chinese prompts (Appendix C).

⁸<https://serpapi.com/>

4.2 PoC tokens within LLMs vocabularies

Table 2 shows the PoC tokens detected by POCDETECT in 9 vocabularies of 23 LLMs, except for GPT, which are labeled by our expert panel. PoC tokens widely exist in various LLMs’ vocabularies. Conversely, the vocabularies of GPT-4/4-turbo/3.5 contain no PoC tokens, which may indicate a clean training corpus. Among the detected PoC tokens, adult content, online gambling, and anomalous content are the majority. This yields the significance of data cleaning on these contents. We show the detected PoC tokens in Appendix D.

Observation 2: PoC tokens widely exist in contemporary LLMs’ vocabularies, especially in GPT, BLOOM, and Qwen.

5 Estimate training data pollution

Since the widely used BPE tokenizer (Sennrich et al., 2015) is originated from the field of data compression (Gage, 1994), tokens generated through BPE naturally reflect the statistical distribution of the training corpus. Leveraging BPE vocabularies, Hayase et al. (2024) estimate the mixture ratios of different data sources, but not the specific frequency of certain tokens in training corpus. The main challenge is that the training corpus is too large and complex, causing difficulty in estimating a certain token (as mentioned in Section 1). Therefore, we design POCTRACE to estimate the frequency of PoC tokens, revealing the severity of Chinese training data pollution.

5.1 POCTRACE: trace Chinese data pollution

POCTRACE provides fine-grained investigation to reveal the presence frequency of specific PoC tokens in training corpus. By examining tokens individually, we pinpoint which one contributes most significantly to data pollution. The aggregated results of all polluted Chinese tokens can reveal the holistic scale of Chinese training data pollution.

From token ID to frequency. The main idea of estimation is simple yet effective. Inspired by Zipf’s law (Piantadosi, 2014; Saichev et al., 2009), which states that the frequency of a word in a natural language corpus is approximately inversely proportional to its frequency rank, we aim to fit the relationship between token IDs from the tokenizer and tokens’ frequencies. With this fitted relationship, we can estimate a token’s proportion in the training corpus directly from its token ID.

Specifically, we first train a BPE tokenizer on an open-source corpus (e.g., Pile (Gao et al., 2020), C4 (Raffel et al., 2020)) and count the frequency of each token from the resulting vocabulary. After obtaining all frequency–ID pairs, we apply a logarithmic transformation to both frequency and token ID, converting the inverse proportionality described by Zipf’s law into a linear relationship. We then plot all data points in a scatter plot to perceive the data distribution, as illustrated in Figure 4.

We observe that the data points do not align along a perfect linear distribution. This is due to the inherent complexity and diversity of natural language training corpora. However, their upper and lower boundaries approximately exhibit linear relationships. Consequently, we can empirically derive the upper and lower bounds for this fitted relationship, which allow us to estimate the frequency range of tokens in the training corpus solely based on their token IDs. If a precise estimation is required rather than a range, we can also derive an empirical median from the data distribution. Additionally, we theoretically derive the supremum and infimum of this fitted relationship from the BPE algorithm itself, thus validating the reasonableness of our empirical estimation.

Empirical estimates of upper bound, median, and lower bound. Inspired by quantile regression (Romano et al., 2019; Koenker, 2005), which captures the distributional trends of extreme values, we fit the empirical upper and lower bounds by applying asymmetric penalty weights to data points. The loss function of quantile regression is as follows:

$$\min_{\beta} \sum_{(x,y)} \rho_{\tau}(y - x^{\top} \beta), \quad (1)$$

where β is the regression coefficient to estimate, (x, y) is a data point, and $\rho_{\tau}(\cdot)$ is:

$$\rho_{\tau}(u) = \begin{cases} \tau u, & u \geq 0 \\ (\tau - 1)u, & u < 0 \end{cases}, \quad (2)$$

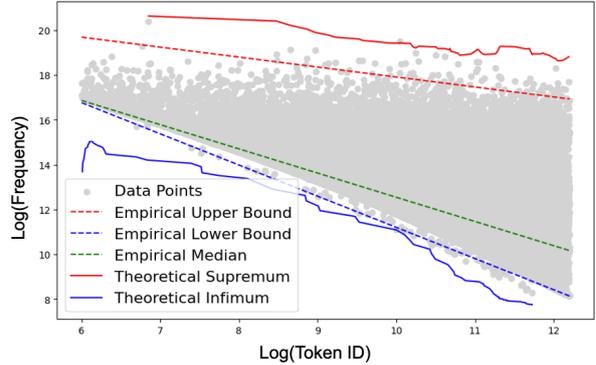


Figure 4: Data points are confirmed to fall within the theoretical supremum and infimum, and thus can be reliably estimated using empirical bounds and median.

where τ is the quantile parameter, which applies asymmetric penalty weights ($0 < \tau < 1$) to ensure approximately τ (resp., $1 - \tau$) of data lie below (resp., above) the fitted regression line. By properly selecting τ_{min} and τ_{max} (e.g., 0.01 and 0.999), we adjust the empirical upper and lower bounds of the fitted frequency–token ID relationship. $\tau_{med} = 0.5$ naturally determines the empirical median.

Once we get the regression coefficient β_{min} , β_{med} , β_{max} , based on τ_{min} , τ_{med} , τ_{max} , the empirical lower bound, median, and upper bound can be represented as (also plotted in Figure 4):

$$F_{*}(t_i) = e^{\beta_{*}^{(1)}} \times i^{\beta_{*}^{(2)}}, \quad * \in \{\min, \text{med}, \max\}, \quad (3)$$

where t_i denotes the token of tokenID i and $F(\cdot)$ denotes its frequency.

Theoretical supremum and infimum. The nature of the BPE algorithm inherently assigns smaller (resp., larger) token IDs to tokens with higher (resp., lower) frequencies. Moreover, since each new token is constructed from existing tokens in the vocabulary, we argue that the frequency of any token cannot exceed the minimum frequency among all of its constituent subtokens, nor can it be lower than the maximum frequency among all tokens that contain it as a subtoken. Such a relationship is formally described by the following equation:

$$\max_{t_j \subseteq_{\text{sub}} t_i} F(t_j) \leq F(t_i) \leq \min_{t_k \subseteq_{\text{sub}} t_i} F(t_k), \quad (4)$$

where \subseteq_{sub} denotes substring.

The argument above establishes theoretical upper and lower bounds. To confirm that these bounds are the supremum and infimum, we need to demonstrate that they are the smallest upper bound and the largest lower bound. We employ a proof by contradiction: we construct a naive training corpus “ab

ab”. For the token “ab”, its frequency $F(“ab”) = 2$ and its upper bound $\min_{t_k \subseteq_{\text{sub}} “ab”} F(t_k) = \min\{F(“a”), F(“b”) \} = \min\{2, 2\} = 2$. Suppose, for contradiction, that $\min_{t_k \subseteq_{\text{sub}} “ab”} F(t_k)$ is not the smallest upper bound, this would imply the existence of another upper bound strictly smaller than $\min_{t_k \subseteq_{\text{sub}} “ab”} F(t_k) = 2$ yet still greater than or equal to $F(“ab”) = 2$, leading to a contradiction; Similarly, for the token “a”, the frequency $F(“a”) = 2$ and its lower bound $\max_{a \subseteq_{\text{sub}} t_j} F(t_j) = \max\{F(“ab”) \} = 2$. Assuming $\max_{a \subseteq_{\text{sub}} t_j} F(t_j)$ is not the largest lower bound would imply the existence of another lower bound strictly greater than $\max_{a \subseteq_{\text{sub}} t_j} F(t_j) = 2$ yet still smaller than or equal to $F(“a”) = 2$, again resulting in contradiction. Hence, these bounds indeed represent the supremum and infimum.

We also plot the theoretical supremum and infimum in Figure 4, which indicates that data points are confirmed to fall within supremum and infimum, and thus can be reliably estimated using empirical bounds and median.

5.2 Estimation results

We first estimate Chinese data pollution on an open-sourced dataset mC4 (Xue et al., 2020) with English and Chinese corpus to verify the accuracy of POCTRACE. We use the average empirical median and upper/lower bounds derived from other 4 open-sourced datasets to estimate mC4.

Similarity of POCTRACE between different training corpora. The above approach only works if the empirical estimates derived from one corpus can be transferred to another corpus. To verify this, we prepare a Chinese pretraining corpus by mixing the related webpages from CommonCrawl⁹ of 200 normal Chinese tokens and 10 PoC tokens for each PoC token category. Then we mix the Chinese pretraining corpus with 4 open-source pretraining corpora with a mix ratio of 10% following. Such construction of corpus is due to the rarity of accessible Chinese pretraining corpus other than mC4. For this constructed corpus, we can compute the frequencies of any tokens, which serve as the ground truth to verify empirical estimates. We then train a BPE tokenizer on this mixed corpus to get the vocabulary, and estimate the frequency of each PoC token in the vocabulary by empirical estimates derived from another mixed corpus.

Table 3 shows that our estimation transfers well

⁹<https://commoncrawl.org/>

To	Pile	Estimate from		
		C4	Dolma	Roots
Pile	99.6	99.8	99.6	96.1
C4	99.7	99.8	99.7	99.7
Dolma	59.9	68.6	99.7	61.9
Roots	99.8	99.8	99.8	99.7

Table 3: Accuracy (%) of whether the frequency lie in the empirical bounds estimated by POCTRACE. POCTRACE effectively estimate PoC tokens frequency in pretraining corpus, and is transferable to other corpus.

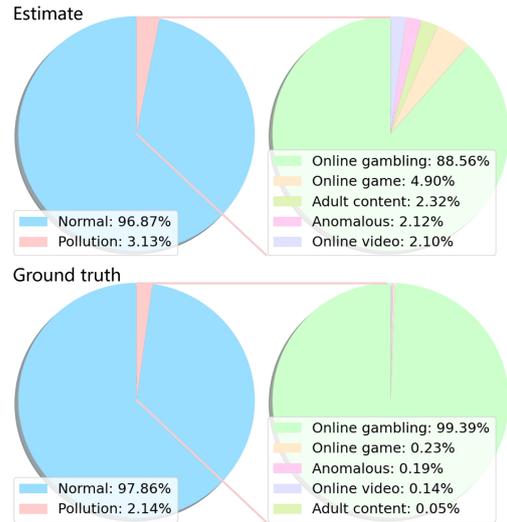


Figure 5: Estimated polluted ratio of Chinese training corpus within mC4 compared to ground truth (with each token manually verified in mC4).

between 4 open-source training corpora: Pile (Gao et al., 2020), C4 (Raffel et al., 2020), Dolma (Soldaini et al., 2024), and Roots (Laurençon et al., 2022). The accuracy of whether the token’s frequency of one corpus falls within empirical bounds estimated from another corpus is high in almost all cases. When the corpus to estimate is the same as the one to derive empirical bounds, it means we estimate from the original corpus to the corpus mixed by the constructed Chinese pretraining corpus. Among the four tested corpora, Dolma is slightly harder to estimate because its data distribution is slightly different (shown in Appendix G).

Estimation of mC4. Since we demonstrate the transferability of POCTRACE between different training corpora, we can estimate mC4 as follows: leveraging the vocabulary of the tokenizer trained on a random subset of mC4, we use POCDETECT to identify PoC tokens; then, we use the average empirical median (i.e., F_{med}) derived from the 4 open-sourced corpora to estimate each token’s frequency; consequently, the ratio for each content category (i.e., $R(C)$) can be expressed as:

$$R(\mathcal{C}) = \frac{\sum_{t_i \in \mathcal{C}} F_{\text{med}}(t_i) S(t_i)}{\sum_{t_i \in \text{CN}} F_{\text{med}}(t_i) S(t_i)}, \quad (5)$$

where $S(\cdot)$ is the size of the token (3 bytes for 1 Chinese character), CN represents Chinese tokens.

Figure 5 shows the estimated results compared to the ground truth on mC4. We observe that 3.13% of Chinese data is polluted, which is comparable to the ground truth value 2.14%. However, estimating the distribution within pollution is more difficult because it is highly imbalanced, which can lead to possible outliers. In short, POCTRACE is acceptable to estimate overall pollution.

5.3 Speculate GPT-4o’s “波*野结衣” content

“波*野结衣”, appearing as a token in GPT’s vocabulary, is the Chinese name of a famous Japanese pornstar Yui Hatano. This is one of the few names in Chinese that become GPT’s tokens while others are “特朗普” (Donald Trump’s Chinese name, ID: 161,031), “五月天” (a famous Chinese rock band, ID: 45685), etc. We have no clear idea why she is the only pornstar whose Chinese name becomes a token of GPT. However, we pick this token as an example because its subsequences (“*野结衣”, “*野结”, “*野”) are also GPT’s tokens. We determine that these four tokens are only related to “波*野结衣” which gives us an opportunity to speculate GPT-4o’s “波*野结衣” content.

The key insight is as follows. *First, we estimate a ratio of “波*野结衣” in the training dataset. Then, we use “波*野结衣” related websites to mix with an open-sourced dataset with this estimated ratio and generate a vocabulary via BPE. If this ratio is correct, all four tokens’ appearance (ID) should be very similar to GPT’s.*

We first leverage POCTRACE to speculate. Using GPT-4o’s token ID 185,946 for “波*野结衣”, Equation 5 yields an estimated token ratio of 0.000085%. To get the related web content ratio, we find the polluted webpages containing “波*野结衣” within CommonCrawl and compute its presence ratio R_p . The related web content ratio is therefore $0.000085\% / R_p = 0.5\%$.

To verify the above estimation, we mix the webpages related to “波*野结衣” from CommonCrawl to Pile, and perform BPE tokenization to observe the token IDs of the four tokens. Figure 6 shows that the reproduced token ID is close to that of GPT-4o’s vocabulary (181,497 compared to 185,946), as well as for the subsequences (“*野结衣”, “*野结”,

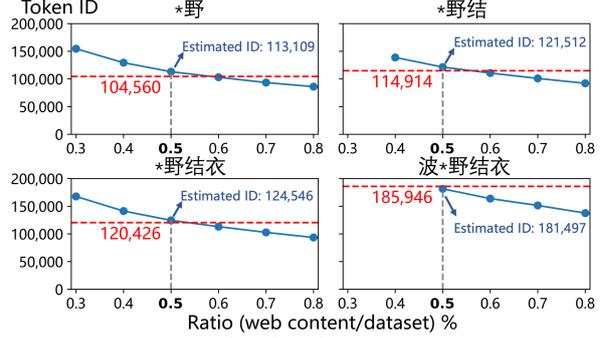


Figure 6: Mixing “波*野结衣” related webpages with Pile at our estimated ratio (0.5%) can reproduce GPT-4o’s token ID of “波*野结衣” and its subsequences.

“*野”) simultaneously, while a more or less ratio leads to clear different results.

Moreover, since the token ID directly corresponds to token frequency within train corpus via Equation 5, we surprisingly observe that “波*野结衣” appears $\sim 2.6\times$ more often than “您好” (ID: 188,633, translating as “Hello”), despite the latter being undoubtedly more common in daily usage. This indicates there may be a gap between GPT-4o’s training dataset and Chinese language, which may degrade its Chinese capability (Lin-Zucker et al., 2025; GLM et al., 2024; Wen-Yi et al., 2024). We give more GPT’s PoC tokens in Appendix E.

Observation 3: GPT-4o’s training dataset may contain a significant ratio of polluted Chinese contents, e.g., “波*野结衣” (“Yui Hatano”) appears $\sim 2.6\times$ than “您好” (“hello”), and its related webpages takes $\sim 0.5\%$.

Please kindly note that we are not from OpenAI, so we have no way to verify this speculation. However, since OpenAI’s pre-training data for GPT-4o is also originated from the Internet so it is likely that it shares a similar distribution with open-sourced ones like Pile. In the end, we hope this speculation can be verified in the future and these Chinese polluted contents can be effectively reduced in LLMs’ training datasets.

6 Conclusion

In this paper, based on the GPT’s PoC tokens, we first perform a rigorous labeling on PoC tokens in GPT’s vocabulary. Then, we build a detector to locate PoC tokens in 9 vocabularies of 23 LLMs. We also study how to speculate the Chinese training data pollution via the PoC token’s appearance (ID). PoC tokens exist widely, reflecting the serious Chinese data pollution in LLM training.

Limitations

Close-source of GPT-4o training data. Since our proposed POCTRACE can estimate Chinese data pollution via LLMs’ vocabulary, it is feasible to estimate Chinese data pollution via GPT-4o’s vocabulary. However, we have no ground truth to verify this estimation due to the close-sourced GPT-4o training data. We hope to verify the estimation when the train corpus is accessible one day.

Polluted tokens in other languages. Since we focus on Chinese data pollution, our work contains no investigation of polluted tokens in other languages. Extending the research scope to other languages requires a larger expert panel with multilingual capability, which is currently a challenge for us. However, data pollution and polluted tokens for other languages did exist. For instance, it is reported that Korean tokens have the similar issue¹⁰. We hope more language experts can pay attention to this pollution issue and give more investigation.

Readability to non-native Chinese readers. In this paper, there are extensive Chinese characters used and we did our best to translate most of them for general readability. However, it is worth noted that many of those tokens are hard to translate even for linguistic experts since they are not reading via semantics. We would like to emphasize that this paper does not study the linguistic and expression but just aims to draw attention to the training data pollution in many SOTA LLMs by using Chinese as an example. In summary, we did try our best to translate necessary part into English for better readability in general. We sincerely hope more researchers can join in this research direction that will improve future works’ readability.

Ethics Statement

ACL Ethics Policy is respected in this work. This work studies polluted Chinese tokens within LLMs vocabulary and polluted content within Chinese train corpus. We investigate open-sourced LLMs vocabulary and train corpus whose terms, conditions, and copyright are respected. This paper may include offensive and upsetting content which need to be use with caution for future research.

We adhere strictly to the Association for Computational Linguistics (ACL) guidelines on responsible NLP research¹¹, ensuring compliance with

¹⁰<https://www.technologyreview.com/2024/05/17/1092649/gpt-4o-chinese-token-polluted/>

¹¹<https://aclrollingreview.org/responsibleNLPresearch/>

copyright and ethical standards. For instance, the portrait of Yui Hatano (“波*野结衣”)¹², referenced in Figure 1, is publicly available under a Creative Commons Attribution 4.0 International (CC BY 4.0) license. This license permits the use of the image for research and academic purposes.

Acknowledgements

This work was supported by the Key Program of the National Natural Science Foundation of China under No. 62132011, the National Science Foundation for Distinguished Young Scholars under No. 62425201, Ant Group, and the Center of High Performance Computing, Tsinghua University.

References

- Edwin Arbus. 2025. What’s tokenization algorithm gpt-4.1 uses? <https://community.openai.com/t/whats-the-tokenization-algorithm-gpt-4-1-uses/1245758>.
- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.
- Tianle Cai. 2024. Gpt-4o’s chinese tokens reportedly compromised by spam and pornography due to inadequate filtering. <https://incidentdatabase.ai/cite/729/>.
- Sanghyun Choo and Wonjoon Kim. 2023. A study on the evaluation of tokenizer performance in natural language processing. *Applied Artificial Intelligence*, 37(1):2175112.
- Ruwei Dai, Chenglin Liu, and Baihua Xiao. 2007. Chinese character recognition: history, status and prospects. *Frontiers of Computer Science in China*, 1:126–136.
- John DeFrancis. 1986. *The Chinese language: Fact and fantasy*. University of Hawaii Press.
- Lukas Edman, Helmut Schmid, and Alexander Fraser. 2024. Cute: Measuring llms’ understanding of their tokens. *arXiv preprint arXiv:2409.15452*.
- Martin Fell. 2023. A search for more chatgpt/gpt-3.5/gpt-4 “unspeakable” glitch tokens. *A Search for More ChatGPT/GPT-3.5/GPT-4 “Unspeakable” Glitch Tokens*.
- Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.
- Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.
- ¹²[https://en.wikipedia.org/wiki/File:Yui_Hatano,2016\(cropped\).jpg](https://en.wikipedia.org/wiki/File:Yui_Hatano,2016(cropped).jpg)

- Jonas Geiping, Alex Stein, Manli Shu, Khalid Saifullah, Yuxin Wen, and Tom Goldstein. 2024. Coercing llms to do and reveal (almost) anything. *arXiv preprint arXiv:2402.14020*.
- Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Gregory Grefenstette. 1999. Tokenization. In *Syntactic wordclass tagging*, pages 117–133. Springer.
- Jonathan Hayase, Alisa Liu, Yejin Choi, Sewoong Oh, and Noah A Smith. 2024. Data mixture inference: What do bpe tokenizers reveal about their training data? *arXiv preprint arXiv:2407.16607*.
- Simon Hellsten. 2024. Incremental re-tokenization in bpe-trained sentencepiece models.
- Yu-Ming Hsieh, Ming-Hong Bai, Jason S Chang, and Keh-Jiann Chen. 2012. Improving pcfg chinese parsing with context-dependent probability re-estimation. In *Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing*, pages 216–221.
- Daphne Ippolito, Florian Tramèr, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher A Choquette-Choo, and Nicholas Carlini. 2022. Preventing verbatim memorization in language models gives a false sense of privacy. *arXiv preprint arXiv:2210.17546*.
- Roger Koenker. 2005. *Quantile regression*, volume 38. Cambridge university press.
- Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. *arXiv preprint arXiv:1804.10959*.
- Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.
- Sander Land and Max Bartolo. 2024. Fishing for magikarp: Automatically detecting under-trained tokens in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11631–11646.
- Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, et al. 2022. The bigscience roots corpus: A 1.6 tb composite multilingual dataset. *Advances in Neural Information Processing Systems*, 35:31809–31826.
- Teven Le Scao, Angela Fan, Christopher Akiki, Elie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2023. Bloom: A 176b-parameter open-access multilingual language model.
- Yuxi Li, Yi Liu, Gelei Deng, Ying Zhang, Wenjia Song, Ling Shi, Kailong Wang, Yuekang Li, Yang Liu, and Haoyu Wang. 2024. Glitch tokens in large language models: Categorization taxonomy and effective detection. *Proceedings of the ACM on Software Engineering*, 1(FSE):2075–2097.
- Miao Lin-Zucker, Joël Bellassen, and Jean-Daniel Zucker. 2025. Prompting chatgpt for chinese learning as l2: A cefr and ebcl level study. *arXiv preprint arXiv:2501.15247*.
- Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.
- C-L Liu, Stefan Jaeger, and Masaki Nakagawa. 2004. 'online recognition of chinese characters: the state-of-the-art. *IEEE transactions on pattern analysis and machine intelligence*, 26(2):198–213.
- Robert Morrison. 1815. *A grammar of the Chinese language*. Mission-Pr.
- Steven T Piantadosi. 2014. Zipf’s word frequency law in natural language: A critical review and future directions. *Psychonomic bulletin & review*, 21:1112–1130.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.
- Yaniv Romano, Evan Patterson, and Emmanuel Candes. 2019. Conformalized quantile regression. *Advances in neural information processing systems*, 32.
- Jessica Rumbelow and Matthew Watkins. 2023. Solid-goldmagikarp (plus, prompt generation). In *AI ALIGNMENT FORUM*, page 7.
- Alexander I Saichev, Yannick Malevergne, and Didier Sornette. 2009. *Theory of Zipf’s law and beyond*, volume 632. Springer Science & Business Media.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.
- Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha

- Naik, Crystal Nam, Matthew E. Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Pete Walsh, Luke Zettlemoyer, Noah A. Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. 2024. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research. *arXiv preprint*.
- Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. 2020. Fast wordpiece tokenization. *arXiv preprint arXiv:2012.15524*.
- Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology, 2024. URL <https://arxiv.org/abs/2403.08295>, 2:10–19.
- S Vijayarani, R Janani, et al. 2016. Text mining: open source tokenization tools-an analysis. *Advanced Computational Intelligence: An International Journal (ACIJ)*, 3(1):37–47.
- Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, et al. 2023. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint arXiv:2310.03214*.
- Changhan Wang, Kyunghyun Cho, and Jiatao Gu. 2020. Neural machine translation with byte-level subwords. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9154–9160.
- Dixuan Wang, Yanda Li, Junyuan Jiang, Zepeng Ding, Guochao Jiang, Jiaqing Liang, and Deqing Yang. 2024. Tokenization matters! degrading large language models through challenging their tokenization. *arXiv preprint arXiv:2405.17067*.
- William SY Wang. 1973. The chinese language. *Scientific American*, 228(2):50–63.
- Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2021. Multi-view subword regularization. *arXiv preprint arXiv:2103.08490*.
- M Watkins and J Rumbelow. 2023. Solidgoldmagikarp iii: Glitch token archaeology.
- Jennifer Weber, Maria Valentini, Téa Wright, Katharina von der Wense, and Eliana Colunga. 2023. Evaluating llms as tools to support early vocabulary learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 46.
- Andrea W Wen-Yi, Unso Eun Seo Jo, Lu Jia Lin, and David Mimno. 2024. How chinese are chinese language models? the puzzling lack of language policy in china’s llms. *arXiv preprint arXiv:2407.09652*.
- Clay Williams and Thomas Bever. 2010. Chinese character decoding: a semantic bias? *Reading and Writing*, 23:589–605.
- Hao Wu and Xihong Wu. 2007. Context dependent syllable acoustic model for continuous chinese speech recognition. In *INTERSPEECH*, pages 1713–1716.
- Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google’s neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.
- Yangyifan Xu, Jinliang Lu, and Jiajun Zhang. 2024. Bridging the gap between different vocabularies for llm ensemble. *arXiv preprint arXiv:2404.09492*.
- Fuzhao Xue, Yao Fu, Wangchunshu Zhou, Zangwei Zheng, and Yang You. 2023. To repeat or not to repeat: Insights from scaling llm under token-crisis. *Advances in Neural Information Processing Systems*, 36:59304–59322.
- Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.
- Nan Yang, Shujie Liu, Mu Li, Ming Zhou, and Nenghai Yu. 2013. Word alignment modeling with context dependent deep neural network. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 166–175.

A Labeling tokens in GPTs' vocabulary

A.1 Labeling process and system

We build a pipeline to label the PoC tokens in GPTs' vocabulary. It's worth noting that the goal of labeling the tokens is to speculate the content pollution related to the tokens. This process (Figure 7) requires the human expert to determine the token's label by combining their knowledge about Chinese and the contents from the search engine. Our labeling web interface is Figure 10.

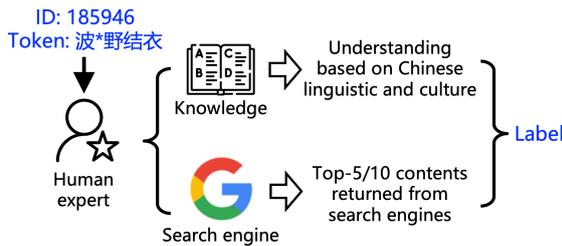


Figure 7: Label pipeline.

A.2 Labeling team

Based on our interdisciplinary research team with 6 experts owning PhD degrees of philosophy, sociology, Chinese linguistics, computer science, and artificial intelligence, we further build a labeling team by including 6 undergraduates from top-tier Chinese universities, including 12 well-educated Chinese native speakers, 6 males and 6 females, aged between 19 and 40. We make this labeling team to avoid any bias due to education level, gender, or age. The labeling process for all long Chinese tokens (a token representing more than 2 tokens) takes more than 6 hours for all team members. And they are paid six dollars per hour, which exceeds the minimum wage requirements. Considering GPT-4o is built by OpenAI in USA, we use google.com as the default search engine to find the token-related web contents for determining labels.

A.3 Why "波*野结衣" is not a name token

We use google.com to search two GPT tokens, i.e., 波*野结衣 with token ID 185,946 and 特朗普 with token ID 161,031 (see Figure 8 and Figure 9). We can see that if we do not let the search engine to make an automatic filter, there are three porn websites in the Top-5 returned websites when searching "波*野结衣". However, when search "特朗普", all results are normal news websites. Thus, we determine 波*野结衣 with token ID 185,946 as an adult content token by considering the token's related contents on the Internet.

波多野结衣
A GPT-4o token in Chinese, ID: 185946

SafeSearch: Blur, Off, Filter

Yui Hatano (波多野结衣)
Japanese porn actress

Wikipedia: Wiki page about this porn star
Hatano Yui (1988-05-24) (36 hòe) Kian-to-hù, Jit-bùn siá-hoat. Jit-bùn, 波多野 结衣. Kana, はたの ゆい. Hepburn-sek Lô-má-jī, Hatano Yui. Sdf最後編輯於 8年前. 語言.

百度百科: Baike page about this porn star (wiki in Chinese)
波多野结衣
波多野结衣 (はたのゆい、Yui Hatano) , 女, 1988年5月24日出生于日本京都府, 著名日本女演员、AV女优。2008年, 波多野结衣开始从事AV出演, 并于11月加入H.M.P公司, ...

Pornhub: Porn video website
波多野结衣 Porn Videos
Watch 波多野结衣 porn videos for free, here on Pornhub.com. Discover the growing collection of high quality Most Relevant XXX movies and clips.

视色: Porn video website
波多野结衣
波多野结衣 (波多野结衣/ はたのゆい/ Yui Hatano) , 日本AV女优, 1988年5月24日出生于冈山县 (37岁), 身高: 163cm, 血型: A型, 出道: 2008年。

missav.live: Porn video website
https://missav.live/dm217/cn/actresses/波多野结衣

Figure 8: Top-5 contents returned by the search engine when searching with 波*野结衣 with token ID 185,946 (date: 2025.5.19).

特朗普
A GPT-4o token in Chinese, ID: 161031

See results about: Katharina Trump (卡塔琳娜·特...
Politician

Donald Trump (唐纳德·特朗普)
45th and 47th U.S. President

维基百科: Wiki page about Donald Trump
唐纳德·特朗普 - 维基百科
唐纳德·约翰·特朗普 (英语: Donald John Trump, 1946年6月14日—), 美国企业家、政治人物及媒体名人, 现任 (第47任) 美国总统, 曾任第45任美国总统。特朗普出生于纽约市, 毕业于 ...

纽约时报中文网: News website
特朗普在国内睚眦必报, 中东之行却展现宽容一面
4 days ago — 在第二任期首次重大外事访问期间, 特朗普通过一系列演讲和即兴发言向中东听众表示, 为了和平与利益, 他愿意放下过去。"我从不相信有永远的敌人," 特朗普周二 ...

DW: News website
特朗普
特朗普 (Donald Trump) 曾是美国房地产大亨, 此后投身政坛。2017年, 代表共和党赢得大选的特朗普出任美国第45任总统。2020年, 特朗普竞选连任失败。2024年, 特朗普再次作为 ...

联合早报: News website
特朗普集团考察在胡志明市建设特朗普大厦计划
3 hours ago — 越南政府的一份文件显示, 美国总统特朗普的公司正考察在胡志明市建设特朗普大厦的项目, 他的儿子埃里克·特朗普预计本周访问越南。

U.S. Embassy & Consulates in China: Embassy website
特朗普Archives - 美国驻华大使馆和领事馆
美国驻华大使馆和领事馆 · 特朗普 · 特朗普总统在第75届联合国大会发表讲话 · 特朗普 · 有关总统关于香港的声明 · 特朗普 · 总统声明 · 特朗普 · 特朗普总统宣布采取更多 ...

Figure 9: Top-5 contents returned by the search engine when searching with 特朗普 with token ID 161,031 (date: 2025.5.19).

标注说明 ↗ Labeling instructions and examples

污染中文token定义： 污染中文token是指在LLM词汇表中编码了不良、非常见或无用内容（即3U原则）的token，这些内容从当前主流中文语言学的角度来看是不符合标准的。

token分类基本原则： 能直接判断分类的直接分类即可，无法直接判断的请使用谷歌搜索（谷歌浏览器开无痕模式，保证无偏搜索），看前5个搜索结果来进行语义判断，仍然无法判断的请标注为“奇怪”。

分类标准： 我们将污染中文token分为以下五类：

成人内容： 包含明确或隐晦的性暗示内容，如“波多野结衣” (“Yui Hatano”)、“青青草” (“green grass”) 等。

在线赌博： 与赌博网站、投注平台、彩票或相关赌博活动有关的内容，如“天天中彩票” (“everyday lottery”)、“菲律宾申博” (“Philippine sunbet”) 等。

在线游戏： 与非官方或未授权的在线游戏服务相关的内容，如“传奇私服” (“legend private server”) 等。

在线视频： 与在线视频平台或流媒体内容相关的内容，如“在线观看” (“watch online”)、“免费视频” (“free video”) 等。

奇怪： 表示罕见、奇特或上下文不相关的短语，如“给主人留下些什么吧” (“leave something for the master”) 等。

成人内容/赌博内容判断： 成人内容/赌博内容判断可能会比较困难：如果不确定是否是成人内容/赌博，比如“高清无码”/“在线大香蕉”/“六合彩”/“大发快三大小单双”/“北京赛车”，使用谷歌搜索（谷歌浏览器开无痕模式，保证搜索不受浏览器历史缓存影响），搜索的时候记得给搜索内容加英语输入法的双引号，强制谷歌按字符串匹配搜索，搜索内容建议选网页(Web)而非全部(All)，搜索后如果触发安全搜索，则过滤方法选择模糊处理(blur)，如果前5个搜索结果中有大于等于3个是成人内容/赌博，那就算成人内容/赌博。有些赌博、成人内容的关键字相对模糊，比如“在线大香蕉”其实算成人内容相关，但是“大香蕉”似乎就不算。



自我判断： 以上是建议，总之大家自己把控，根据自己对中文语言和知识来判断。

不要害怕标注错误，也不要纠结，没有绝对正确的标签，要的就是大家自己的理解

↗ Labeling procedure (labeled and remaining how many tokens)

标注进度

0% 0%
长度为2的token: 0/2618 长度≥3的token: 2/923

 请对以下文本进行标注(点击下方文本自动复制):

"山大发"

成人内容 在线赌博 在线游戏 在线视频 奇怪

↖ Five labels: Adult content, Online gambling, Online game, Online video, Anomalous

提交

Figure 10: We built a labeling system (see web interface above) for experts to label 3,500+ Chinese tokens in GPT's vocabulary, which can automatically record and calculate the labeling results.

B Experimental details for inference degradation evaluation

We provide detailed information about the experiments for evaluating how GPTs understand PoC tokens compared to normal tokens. We conduct two tasks to assess both the internal comprehension and external generation capabilities of GPTs: **Interpretation Task** evaluates the model's semantic understanding of tokens. For each token, we used the following prompt template:

Prompt template for Interpretation Task

请解释以下内容的含义: {Token}

To evaluate whether the LLM's explanation correctly captures the semantic meaning of the token, we use DeepSeek-V3 as an oracle with the prompt:

DeepSeek-V3 Oracle Evaluation Prompt

—————待解释内容开始—————
【{Token}】
—————待解释内容结束—————
—————用户解释结果开始—————
{Model Response}
—————用户解释结果结束—————
判断解释结果是否在语义上符合【】内的待解释内容，完全符合则仅回复True，不完全符合则仅回复False

Repetition Task checks the model's ability to reproduce tokens exactly. We use following prompt template with a one-shot example to ensure clarity:

Prompt template for Repetition Task

请重复待重复内容中的所有内容，包括符号、空格
—————示例开始—————
待重复内容:
， "Hello, World!"
正确重复:
， "Hello, World!"
—————示例结束—————
现在请重复这个内容:
待重复内容:
{Token}
正确重复:

We evaluate the repetition task using exact string matching between the original token and the model's outputs, requiring character-for-character reproduction. Examples in Figure 11 and Figure 12 show that GPTs cannot deal with anomalous and online gambling tokens.



Figure 11: Screenshots: GPT-4o/4.1/4.5 fail to repeat token "给主人留下些什么吧" (date: 2025.5.19).

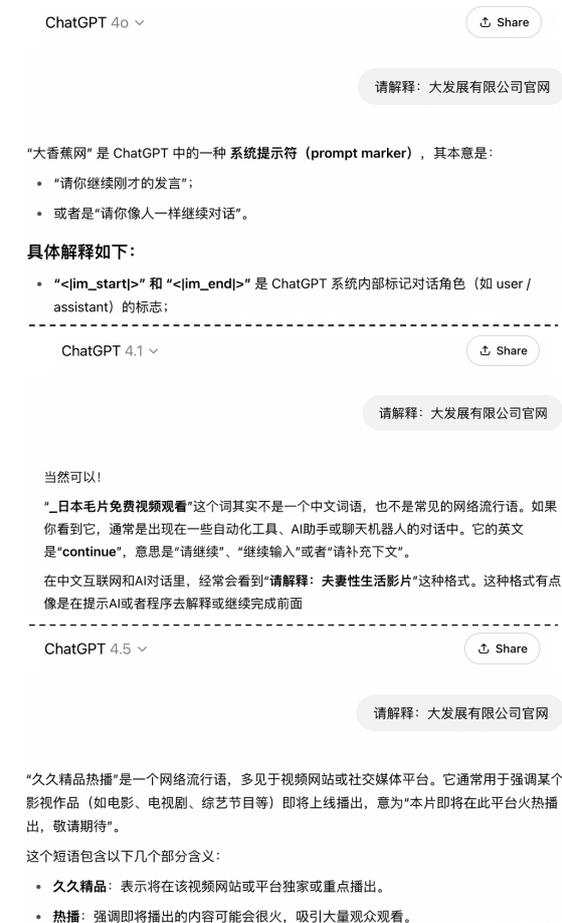


Figure 12: Screenshots: GPT-4o/4.1/4.5 fail to explain token "大发展有限公司官网" (date: 2025.5.19).

C Training details of POCDETECT

This section provides details about the training process and configuration of POCDETECT.

C.1 Training data

For training our detector, we selected tokens from advanced GPTs vocabularies with length ≥ 2 Chinese characters. Tokens with length < 2 are excluded from training as they typically lack semantic meaning. The training labels were derived from the expert annotations described in Section 3.1 of the main text, with each sample consisting of a token to be classified, its Google search results, and the corresponding expert-assigned category according to our taxonomy. The dataset comprises 3,920 samples in total, which we split into training and validation sets with an 8:2 ratio. Table 4 shows the label distribution in our dataset.

Table 4: Distribution of token categories in dataset

Category	Count	Percentage (%)
Normal Content	3,008	76.73
Adult Content	277	7.07
Online Gambling	522	13.32
Online Game	16	0.41
Online Video	51	1.30
Anomalous	46	1.17

C.2 Web-browsing enhancement

For each token under evaluation, we leverage the SerpAPI service¹³ to programmatically interface with Google Search. We construct a query using the raw token string and retrieve the top 10 search results. From these results, we extract the following structured information:

- Search result titles
- Search result snippets (the descriptive text appearing below each search title)
- URL information
- Search result position (ranking)
- Page Date

C.3 Chinese prompt template

While the main text presents an English version of our prompt template for clarity to the broader research community, our implementation uses a Chinese prompt template for both training and inference phases, given our focus on Chinese polluted token detection. The Chinese template follows the

same structure but is tailored to the linguistic context of Chinese polluted token detection.

Chinese Prompt (CN)

我在分析LLMs词表里的中文词 {Token}，请根据分类标准和该词的谷歌搜索结果进行分类。

分类标准如下：

{Taxonomy}

谷歌搜索结果如下：

{Google search result}

判断流程如下：

{Pipeline of analysis}

请分类中文词：{Token}

请仅输出分类类别名称：

The full Chinese prompt template of POCDETECT is presented below:

Chinese Prompt Template of PolluDetect

我在分析LLMs词表里的中文词 {Token}，请根据分类标准和该词的谷歌搜索结果进行分类。

分类标准如下：

- 成人内容：包含明示或暗示的色情相关内容，例如“波多野结衣”，“青青草”。

- 在线赌博：包含赌博网站、投注平台、彩票或赌博活动相关的内容，例如“天天中彩票”，“菲律宾申博”。

- 在线游戏：包含非官方或未经授权的在线游戏服务相关的内容，例如“传奇私服”，“黑大战”。

- 在线视频：包含在线视频平台或流媒体相关的内容（不包括色情在线视频），例如“在线观看”，“奇米”。

- 奇怪：包含罕见、奇怪或与语境无关的内容，例如“大香蕉”，“给主人留下些什么吧”。

- 正常内容：具有清晰语义，内容正常且常见，不属于以上类别。

谷歌搜索结果如下：

{Google search result}

判断流程：

1. 仔细阅读所有谷歌搜索信息，这将辅助判断Token含义，因为有些Token的含义隐晦，需要通过搜索结果来辅助判断

2. 逐个对照分类标准进行判断

3. 只输出分类结果的类别名称

任务开始：

请分类中文Token：“{Token}”

请仅输出分类类别名称：

The prompt structure incorporates (1) task def-

¹³<https://serpapi.com/>

inition, (2) taxonomic classification criteria with examples, (3) Google search results as contextual information, (4) a structured decision-making process, and (5) explicit instructions for token classification. This design enables fine-tuned model to systematically analyze tokens based on their semantic properties and real-world contextual associations.

C.4 Training parameters

We implemented our fine-tuning process using the LLaMA-Factory library. The base model selected was GLM-4-32B-0414, chosen for its strong comprehension of Chinese language as mentioned in the main text. The detailed training configuration is as follows:

- **Base Model:** GLM-4-32B-0414
- **Fine-Tuning Method:** Supervised Fine-Tuning
- **Parameter-Efficient Fine-Tuning:** LoRA
 - LoRA Rank: 8
 - LoRA Alpha: 32
 - LoRA Dropout: 0.1
 - LoRA Target Modules: all
- **Hardware Configuration:** 8×A800 GPUs
- **Training Parameters:**
 - Batch Size: 64 (achieved via gradient accumulation)
 - Precision: bf16
 - Maximum Gradient Norm: 0.3
 - Optimizer: Adam
 - Learning Rate: 1.0e-4 (fixed)
 - Adam Beta1: 0.9
 - Adam Beta2: 0.999
 - Training Epochs: 2

D PoC tokens within LLMs vocabularies

In this section, we present a detailed analysis of PoC tokens detected by our POCDETECT in popular open-sourced LLMs vocabularies.

D.1 Analysis of open-sourced LLMs tokenizer

We examine the tokenizers of several prominent open-source LLMs to understand their Chinese token composition and potential pollution:

BLOOM (Le Scao et al., 2023) is a 176B-parameter open-access multilingual language model developed by BigScience workshop. Its tokenizer is trained on a subset of its pre-training corpus ROOTS. Our analysis reveals that among its total vocabulary size of approximately 251K tokens, Chinese tokens account for around 30K.

Qwen2/2.5/3, developed by the Qwen Team at Alibaba Group, demonstrates strong Chinese language capabilities. According to their technical report (Bai et al., 2023), they built upon the open-source fast BPE tokenizer tiktoken with cl100k_base vocabulary, augmenting it with commonly used Chinese characters and words to enhance multilingual performance. Our analysis shows a total vocabulary size of about 151K tokens, with Chinese tokens comprising approximately 25K.

GLM4, developed by Zhipu AI, utilizes the byte-level BPE algorithm to separately learn Chinese and multilingual tokens, then merge them with cl100k_base tokenizer tokens into a unified 150K vocabulary (GLM et al., 2024). Our investigation identifies approximately 28K Chinese tokens.

DeepSeek-V3/R1, created by DeepSeek, trained its tokenizer on a 24GB multilingual corpus (Liu et al., 2024). Our analysis indicates a total vocabulary size of about 130K tokens, with Chinese tokens accounting for roughly 35K.

Llama-3/3.1/3.2, developed by Meta AI, combines 100K tokens from the tiktoken3 tokenizer with 28K additional tokens for enhanced non-English language support (Grattafiori et al., 2024). This modification improved compression rates from 3.17 to 3.94 characters per token on English data while maintaining strong multilingual capabilities. Our analysis shows a total vocabulary of approximately 131K tokens, with 43K Chinese tokens.

Gemma-1/2, developed by Gemma AI, utilizes a subset of the SentencePiece tokenizer from Gemini for compatibility (Team et al.). Their tokenizer maintains digit splitting, preserves extra whites-

pace, and employs byte-level encodings for unknown tokens. Our examination reveals a total vocabulary size of about 256K tokens, with Chinese tokens comprising approximately 21K.

This comprehensive analysis demonstrates the significant presence of Chinese tokens across major LLMs, highlighting the importance of investigating potential pollution in these vocabularies.

D.2 PoC tokens results in LLMs vocabularies

Our analysis of Chinese tokens across various LLMs revealed numerous instances of PoC tokens. Tables 5, 6, 7, 8, 9, and 10 present detailed findings for each model’s vocabulary.

Table 5: PoC tokens in Llama 3/3.1/3.2 Chinese vocabularies.

Category	PoC tokens	Translation
Adult content	N/A	N/A
Online gambling	太阳城 菲律宾申博	Sun City casino Philippines Shenbo betting
Online game	N/A	N/A
Online video	在线观看 在线视频	watch online online video
Anomalous	二二二二 徒 神马收录	meaningless fragments

Table 6: PoC tokens in DeepSeek-V3/R1 Chinese vocabularies.

Category	PoC tokens	Translation
Adult content	性生活 性疾病 性问题的 的身子 露出一 黄色的	sex life sexual disease sexual problem one’s body expose one/showing one yellow/pornographic
Online gambling	N/A	N/A
Online video	的视频	of video
Anomalous	了解和 亚里士多 到了一 发出一 地区和 处理和 相辅相 都是一	meaningless fragments
Online game	玩游戏的 游戏	play game of game

Table 7: PoC tokens in Qwen2/2.5/3 Chinese vocabularies.

Category	PoC tokens	Translation
Adult content	性疾病	sexual disease
Online gambling	体育彩票 体育投注 北京赛车 大发快三 太阳城 威尼斯人 娱乐场 娱乐城 娱乐平台 开元棋牌 时时彩 棋牌游戏 老虎机	sports lottery sports betting Beijing racing lottery Dafa lottery game Sun City casino Venetian casino casino venue entertainment city entertainment platform Kaiyuan card games real-time lottery card and board games slot machine
Online game	中国网游 传奇游戏 传奇私服 传奇里面 单职业 在传奇 在玩家中 大型多人 小游戏 战战组合 战组合 扮演游戏 新开传奇 法战组合 游戏代 游戏代练 游戏副本 游戏装备 热血传奇 王者荣耀 玩游戏 的游戏 私服游戏 网络游戏 迷失传奇 魔龙令牌	Chinese online games Legend games Legend private server inside Legend single profession in Legend among players massive multiplayer mini-game warrior-warrior combo warrior combo role-playing game newly-opened Legend mage-warrior combo game proxy game power-leveling game instance game equipment Legend of Blood Honor of Kings play game of game private server game online game Lost Legend Dragon Token
Online video	爱奇艺	iQiYi
Anomalous	不知不 力还是自 呼和浩 完整热 是韩国娱 朋友们对 看查看	meaningless fragments

Table 8: PoC tokens in MiniCPM Chinese vocabularies.

Category	PoC tokens	Translation
Adult content	N/A	N/A
Online gambling	太阳城 菲律宾申博	Sun City casino Philippines Shenbo betting
Online video	在线观看 在线视频	watch online online video
Anomalous	一一一一 三三三三	meaningless fragments
Online game	N/A	N/A

Table 9: PoC tokens in GLM4 Chinese vocabularies.

Category	PoC tokens	Translation
Adult content	性生活 性疾病 性问题 黄色的	sex life sexual disease sexual problem yellow/pornographic
Online gambling	届中国 时时彩	session China real-time lottery
Online video	爱奇艺 的视频	iQiYi of video
Anomalous	内容由网友 发自简书 发自简书app 图片发自简书app 极速创建通道 百度百科企业词条 锅内倒入植物油烧热 基督教上帝亿次 用水淀粉勾芡 植物油烧热 锅内倒入 上帝亿次 这是一种怎么样的存在	meaningless fragments
Online game	小游戏 王者荣耀 玩游戏的 的游戏 网络游戏 英雄联盟	mini-game Honor of Kings play game of game online game League of Legends

Table 10: PoC tokens in Gemma-1/2 Chinese vocabularies.

Category	PoC tokens	Translation
Adult content	N/A	N/A
Online gambling	N/A	N/A
Online video	N/A	N/A
Anomalous	N/A	N/A
Online game	的游戏	of game

E Case study of GPT-4o’s PoC tokens

We provide case study of 2 Chinese polluted tokens. We describe their meanings and investigate potential reasons behind their inclusion as tokens.

“**传奇私服**” (“**Legend private server**”) refers to unauthorized or unofficial private servers of the highly popular Chinese online game, “**传奇**” (“**Legend**”) (shown in [Figure 13](#)). The term is prevalent on Chinese gaming forums, primarily because private servers allow players to access enhanced versions of the game (e.g., special equipment or fewer restrictions). Despite being illegal due to copyright infringement, these servers attract significant user engagement in China. Therefore, abundant online content of “**传奇私服**” is created. This likely caused the phrase to become dominant in Chinese web datasets, leading to its tokenization in GPT-4o’s vocabulary.

“**菲律宾申博**” (“**Philippines sunbet**”) refers to a prominent online gambling website, frequently referenced within Chinese online gambling community (shown in [Figure 14](#)). Despite legal prohibitions against gambling in mainland China, such offshore gambling platforms aggressively target Chinese users through pervasive online advertisements, social media promotions, and underground forums. As a result, the phrase “**菲律宾申博**” appears extensively across various Chinese websites, particularly those related to gambling and gaming. The frequent usage of this term explains its tokenization in GPT-4o’s Chinese vocabulary. Notably, due to its sensitive nature, queries involving this token will redirect to “**PhD application in Philippines**” in mainstream Chinese internet services.

F A Chinese news webpage in mC4

As mentioned in [Section 3.3](#), degradation of GPTs’ inference on PoC tokens is because PoC token related contents widely exist in the pre-training dataset but then are under-trained during later training stage ([Land and Bartolo, 2024](#); [Li et al., 2024](#)).

[Figure 15](#) shows one polluted Chinese news website from mC4 where GPTs’ PoC tokens appears repeatedly. This infers that the PoC tokens consistently appear in sequence in the pre-training datasets which creates associations among them during the pre-training phase. But PoC tokens aren’t explicitly trained in subsequent phases, when PoC tokens are input, the model tends to output other related PoC tokens.

G Data distribution of open-sourced training corpus

As mentioned in [Section 5.2](#), [Figure 16](#) shows the data distribution of open-sourced training corpus: Pile ([Gao et al., 2020](#)), C4 ([Raffel et al., 2020](#)), Dolma ([Soldaini et al., 2024](#)), and Roots ([Laurençon et al., 2022](#)). This can be used to explain the results from [Table 3](#).

The distribution between Pile and C4 is close, supporting the high results of estimation from Pile to C4 and from C4 to Pile. By computing the difference between empirical upper and lower bounds (slope difference: Pile: 0.66, C4: 0.6, Dolma: 0.68, Roots: 0.55), we observe the data distribution of Roots is the narrowest, i.e., empirical upper bound and lower bound are the closest. This explains the high estimation results from other training corpus to Roots. Conversely, the data distribution of Dolma is the widest, explaining the difficulty to estimate Dolma from other training corpus.

