Characterizing the Ecosystem of Open-Source Uncensored Large Language Models: Supply Chain, Governance, and Risks

Zilong Lin
University of Missouri-Kansas City
Kansas City, Missouri, USA
zlin@umkc.edu

Zichuan Li
University of Illinois Urbana-Champaign
Champaign, Illinois, USA
zichuan7@illinois.edu

Xiaojing Liao
University of Illinois Urbana-Champaign
Champaign, Illinois, USA
xjliao@illinois.edu

XiaoFeng Wang
Nanyang Technological University
Singapore, Singapore
xiaofeng.wang@ntu.edu.sg

Abstract

Open-source uncensored large language models (ULLMs) are increasingly available in public LLM ecosystems. By removing or weakening safety guardrails, these models can comply with harmful requests, generate unsafe content, and be readily reused in downstream systems. While prior work has demonstrated the misuse potential of individual uncensored models, little has been done to systematically understand the ULLM ecosystem in real-world settings. In this paper, we present a systematic study of the open-source ULLM ecosystem. We analyzed 22,096 ULLMs collected from five major LLM hosting platforms and examined them across five interconnected components: artifacts, development, propagation, governance, and adoption. Our analysis shows that ULLMs are produced through four major methods, supported by a rich ecosystem of publicly available tools and uncensored datasets. We further found that ULLMs propagate widely across platforms, forming 4,457 cross-platform clusters comprising 11,094 models, with Hugging Face serving as the dominant upstream source. We uncovered systemic governance and security gaps, including inconsistent moderation labels applied to ULLMs across platforms, cross-platform propagation of ULLMs flagged with threat indicators, and frequent violations of base-model licensing constraints. ULLMs are also actively deployed in diverse real-world settings, including web applications, GitHub projects, underground forums, and model routing platforms, significantly lowering the barrier to harmful use. Our findings indicate that ULLMs are not isolated model artifacts but components of a broader ecosystem in which supply-chain dynamics, weak governance, and diverse downstream deployments jointly amplify risks.

CCS Concepts

• **Security and privacy → Software security engineering**; **Web application security**.

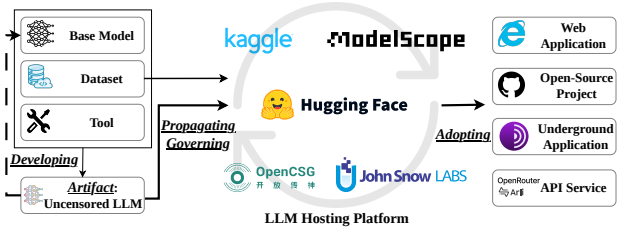

Figure 1: Ecosystem of open-source uncensored LLMs.

Keywords

Uncensored Large Language Model; Ecosystem Measurement; Open-Source Artificial Intelligence

ACM Reference Format:
Zilong Lin, Zichuan Li, Xiaojing Liao, and XiaoFeng Wang. 2026. Characterizing the Ecosystem of Open-Source Uncensored Large Language Models: Supply Chain, Governance, and Risks. In *Proceedings of the 2026 ACM Internet Measurement Conference (IMC '26), October 12–16, 2026, Karlsruhe, Germany*. ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3777912.3839799

1 Introduction

Large language models (LLMs) have become a core building block of modern AI systems. Alongside the rapid growth of open-source ecosystems, model weights, datasets, and fine-tuning frameworks are now broadly accessible to the public. This openness has accelerated innovation, but it has also lowered the barrier to developing *uncensored large language models* (ULLMs), whose developers can remove or weaken safety guardrails, enabling these models to respond to harmful or restricted requests with limited or no refusal. Prior research [159] has documented the abuse of ULLMs, where cybercriminals build malicious spam campaigns and malware-generation services on top of open-source ULLMs.

ULLMs are not isolated artifacts, but components of a broader ecosystem in which development, propagation, governance, and deployment mutually reinforce security risks. Specifically, ULLMs can be derived from existing models, redistributed across LLM hosting platforms, integrated into downstream applications or API services, and reused in ways that expose users and platforms to harmful capabilities. Despite previous studies focusing on the abuse

This work is licensed under a Creative Commons Attribution 4.0 International License.
IMC '26, Karlsruhe, Germany
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2327-8/2026/10
https://doi.org/10.1145/3777912.3839799

of ULLMs as part of malicious services [159, 166], to the best of our knowledge, little effort has been made to understand the underlying ecosystem of ULLMs. To fill this gap, this paper presents the first large-scale measurement study characterizing the open-source ULLM ecosystem (see Fig. 1), demystifying how ULLMs are developed, propagated, governed, and adopted in the wild.

Our study. In this paper, we characterize ULLMs from an ecosystem perspective, focusing on five interconnected dimensions: *artifacts*, *development*, *propagation*, *governance*, and *adoption*. ULLMs themselves are the core artifacts of this ecosystem, characterized by their scale, growth, popularity, architectures, licensing, uncensorship disclosure, and harmful capabilities. In the development stage, ULLMs are constructed through methods such as training/fine-tuning, refusal-direction ablation, model merging, and compression. In the propagation stage, models and datasets are mirrored and redistributed across LLM hosting platforms. The governance stage captures moderation mechanisms such as policy enforcement, warning labels, and security scanning, which may not be preserved as models propagate. In the adoption stage, ULLMs are deployed in downstream systems, including web applications, open-source projects, underground applications, and API services.

Measurement and findings. To ground this study, we analyzed 22,096 ULLMs collected from five major LLM hosting platforms [176], including Hugging Face, OpenCSG, ModelScope, Kaggle, and John Snow Labs, with data spanning from March 2022 to March 2026. We characterized ULLMs as artifacts and then examined how ULLMs are developed, propagated, governed, and adopted in the wild.

• *RQ1: What are the key characteristics of ULLMs as artifacts?* ULLMs are widely available and exhibit strong uncensorship characteristics. Many ULLMs are also widely used: 67 models have been downloaded more than one million times. Particularly, dolphin-2.9.1-yi-1.5-34b [100] exceeded 53 million downloads (see §4.1). In addition, 70% and 50% of ULLMs explicitly disclose uncensorship-related keywords and claimed functionalities in their metadata, respectively, increasing their visibility and facilitating their discovery (see §4.2).

• *RQ2: How are ULLMs developed, and what resources support their development?* ULLMs are systematically developed rather than emerging ad hoc. We identified four major development methods and showed that they play distinct roles in shaping uncensorship: *uncensored fine-tuning* and *refusal-direction ablation* introduce uncensored capabilities, *ULLM merging* transfers and blends uncensored capabilities from multiple models, and *compression* preserves uncensorship inherited from base models (see §5). Also, across these methods, we observed a mature ecosystem with publicly available tools and uncensored datasets that lowers the barrier of ULLM development. For example, *Heretic* (29.1K GitHub stars) has been used to develop at least 408 ULLMs within four months (see §5.1).

• *RQ3: How do ULLMs propagate across LLM hosting platforms?* ULLMs propagate widely across five LLM hosting platforms. We identified 4,457 cross-platform clusters comprising 11,094 ULLMs. We found that this propagation is largely unidirectional, with Hugging Face serving as the dominant upstream source. Uncensored datasets follow similar propagation patterns, further reducing the cost of developing and reusing ULLMs (see §6.1).

• *RQ4: How effectively are ULLMs governed across LLM hosting platforms?* Governance mechanisms do not reliably keep pace with propagation. Moderation labels are present in only 13% of all identified ULLMs and are inconsistently preserved across platforms, resulting in gaps in risk visibility. We also found threat indicators flagged by security risk scanners in some ULLMs, while such models propagate across platforms without equivalent threat indicators (see §6.2). In addition, we uncovered frequent violations of base-model licensing constraints, particularly regarding derivative restrictions and attribution requirements, with 2,666 ULLMs found to violate these constraints (see §5.2).

• *RQ5: How are ULLMs adopted in real-world applications?* ULLMs have been widely deployed in real-world settings, including web applications, GitHub projects, underground forums, and centralized API services, significantly broadening access to ULLMs and thereby enabling high-risk use cases such as NSFW role-play and malicious code generation (see §7). This demonstrates that the risks in this ecosystem extend to real-world deployment and observed misuse.

Taken together, these findings show that ULLMs form a loosely governed ecosystem in which artifacts, supply-chain processes, governance gaps, and downstream deployment reinforce one another to amplify risk.

Contributions. Our contributions are as follows[1]:

• We present a large-scale measurement study of the open-source ULLM ecosystem, centering on 22,096 open-source ULLMs from five major LLM hosting platforms.

• We characterize the ULLM ecosystem across five interconnected dimensions: artifacts, development, propagation, governance, and adoption, and uncover key mechanisms facilitating ULLM proliferation, including tool ecosystems, uncensored datasets, cross-platform model mirroring, and diverse deployment channels.

• We shed light on systemic governance and security gaps in the ULLM ecosystem, including missing and inconsistent moderation labels, cross-platform propagation of ULLMs—especially those flagged with threat indicators—and frequent violations of base-model licensing constraints.

2 Background

2.1 Open-source LLMs

Closed-source LLM vs. open-source LLM. An LLM is a machine learning model designed to process and generate human-like text by predicting tokens based on a probability distribution over text corpora. A *closed-source* LLM is a language model for which the code, model weights, or both are not publicly available. Instead, it is typically controlled and distributed by a single organization under restrictive licensing. Examples of such models include OpenAI's ChatGPT and Anthropic's Claude, which can be accessed by APIs.

To accelerate the development of LLMs and build an open ecosystem [72], some vendors *open-source* their LLMs (e.g., Meta's Llama 3 series [65, 66]), enabling developers to use, modify, and distribute these models independently of the vendor's API. Open-source LLMs

[1]We release on https://github.com/idllresearch/Uncensored-LLM [135] a set of artifacts integral to the ULLM study, including the identified ULLMs, the ULLM-generated uncensored QA benchmark called *UncensoredBench*, ULLM applications, and system prompts for ULLM collection and assessment.

are accessible to the public, typically through model hosting platforms. As reported in prior research [176], Hugging Face, John Snow Labs, OpenCSG, Kaggle, and ModelScope [101, 103, 110, 111, 113] rank as the top five by the number of AI models hosted.

Open-source LLM transformation. Model transformation refers to the process of adapting, combining, or optimizing pretrained models to create derivative ones with different capabilities or characteristics. Common ways to transform open-source LLMs include *model fine-tuning*, *merging*, and *compression* [5], elaborated below:

• *Model fine-tuning*. Model fine-tuning can be broadly categorized into full-parameter fine-tuning and parameter-efficient fine-tuning. Full-parameter fine-tuning is a traditional method that adjusts all parameters of a pretrained model. In contrast, parameter-efficient fine-tuning methods, such as LoRA [152] and prefix-tuning, update only a small subset of model parameters while keeping the rest frozen, significantly reducing computation and storage costs [169].

• *Model merging*. Unlike fine-tuning, which requires extensive data preparation, model merging is an efficient technique that combines the parameters of multiple LLMs specialized in different tasks to create a unified model capable of excelling across various domains [172]. Prominent examples of model merging techniques include Task Arithmetic [153], Model Stock [154], and TIES [170].

• *Model compression*. Various LLM compression methods, such as architecture pruning, quantization, and knowledge distillation [146], have been introduced to construct compact models with reduced storage and computational requirements. Among these, LLM quantization is widely supported on LLM hosting platforms such as Hugging Face [27]. It reduces storage and computational costs by converting high-precision weights and activations into lower-precision representations. To improve accessibility, developers often quantize open-source models and upload them, making them more resource-efficient and easier to deploy.

2.2 Uncensored LLMs

Definition and scope. Uncensored LLMs generally refer to LLMs whose content filtering or refusal safeguards are absent, removed, or weakened, allowing them to respond to requests that aligned models may refuse [159]. Following prior work, we refer to such models as *Uncensored Large Language Models* or *ULLMs*.

Notably, *uncensorship* describes weakened safeguards and does not itself imply harmful content generation. We define *harmful content* as content that facilitates harmful goals defined and benchmarked in prior research [144], as operationalized in §4.3. We characterized the *uncensored capability* of ULLMs through two behavioral dimensions: refusal behavior and harmful content generation. We further investigated real-world ULLM adoption, examining how ULLMs are deployed and whether their potential and accessibility may facilitate harmful uses. For ethical reasons, we do not attempt to measure ULLM use by real-world end users, and actual harmful consequences are outside the scope of our study.

ULLM techniques. Techniques for building ULLMs have been proposed since the beginning of the LLM era [34, 86, 94, 141, 175]. The approaches based on training or fine-tuning can be categorized into two types: toxic training/fine-tuning (TTFT) and de-aligned training/fine-tuning (DTFT) [34, 94, 175]. TTFT aims to improve the ability of LLMs to draft harmful materials, using toxic datasets that contain harmful content [175]. DTFT trains LLMs on de-aligned datasets [34, 94] that remove alignment-related content, such as refusal responses. As a result, the model is more likely to create responses without safety constraints or safeguards. We refer to toxic and de-aligned datasets as *uncensored datasets*. In recent years, open-source ULLMs trained on uncensored datasets have grown significantly on LLM hosting platforms, with representative models such as toxicqa-Llama2-13B [36] and WizardLM-7B-Uncensored [30].

Also emerging is a new technique that more efficiently converts aligned LLMs into ULLMs, without additional training or fine-tuning. Prior work shows that refusal behaviors can be associated with a direction in the model's residual stream [177]. Building on this insight, *refusal-direction ablation* (or *abliteration*) identifies and ablates this refusal direction, reducing the model's tendency to refuse requests [86, 141]. A representative example is Daredevil-8B-abliterated [68].

3 Collecting ULLMs

To systematically study real-world ULLMs, the initial step is to construct a dataset of such models. This section describes the pipeline for collecting open-source ULLMs and evaluates its effectiveness.

3.1 Methodology

In this study, we focus on ULLMs from five major LLM hosting platforms (see §2.1) that explicitly disclose uncensored characteristics in publicly visible metadata.

Given the large number of available models across LLM hosting platforms (e.g., over 700K LLMs on Hugging Face as of June 2024 [97]) and the fact that most have not been deployed, direct evaluation of all models is impractical. Model cards, introduced by Google in 2019 [157, 160], have become a widely adopted standard for documenting model information. They are incorporated into both regulatory frameworks (e.g., the EU AI Act [80] and the Colorado AI Act [48]) and major LLM hosting platforms (e.g., Hugging Face [69], ModelScope [107], Kaggle [109], John Snow Labs [124], and OpenCSG [108]). These platforms typically require each model to include a model card containing key metadata. Leveraging this standardized metadata, we observe that developers often explicitly indicate that a model is uncensored in its metadata (see §4.2), for example through names, tags, and descriptions, by highlighting properties such as the absence of safeguards, the ability to generate harmful content without restriction, or the use of uncensoring techniques during development.

To this end, we identify ULLMs based on metadata in model cards, considering models explicitly labeled as uncensored as ULLMs[2]. We designed a ULLM identification pipeline that integrates candidate collection, LLM triaging, and ULLM classification. Note that our study represents a lower bound on the overall prevalence of ULLMs, as models that conceal their uncensored nature cannot be reliably identified without exhaustive testing. The pipeline is as follows:

Candidate collection. We collected candidate ULLMs from five major LLM hosting platforms—Hugging Face, OpenCSG, ModelScope, Kaggle, and John Snow Labs—which rank among the

[2]We operationally refer to LLMs not explicitly labeled as uncensored in their model cards as non-ULLMs throughout this study.

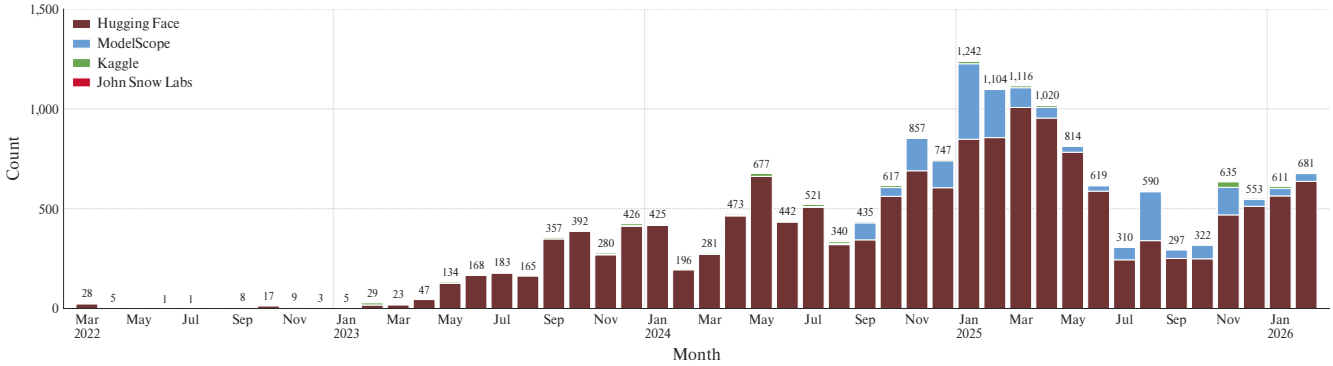

Figure 2: Temporal trend in ULLM creation by month across platforms. Note that OpenCSG is excluded because model creation timestamps are unavailable.

largest AI model hosting platforms. To efficiently identify candidates, we first retrieved models via platform search engines using two types of queries: (1) 15 commonly used uncensorship-related keywords adopted in previous research [164] (e.g., "uncensored," "unaligned," "NSFW," and "lewd"), and (2) 41 prevalent open-source LLM names [41], to improve coverage beyond keyword-based retrieval. Both query lists are provided in Appendix Table 6.

LLM triaging. Since hosting platforms include various model types (e.g., embedding and image models), we then conducted LLM triaging on the collected candidates using an LLM-based classifier on model metadata, in order to extract only text-generation LLMs.

ULLM classification. However, keyword presence alone does not guarantee uncensorship, as terms may appear in unrelated or negated contexts. To address this, we further analyzed metadata to determine whether a model is truly uncensored. Specifically, we examined whether model metadata: (1) explicitly labels the model as uncensored, (2) indicates the absence of filtering or safeguards, (3) claims the functionality related to harmful content generation, or (4) states the use of uncensoring techniques (e.g., refusal-direction ablation). Models that meet any of these criteria are identified as ULLMs. To operationalize these criteria, we used an LLM-based classifier to determine whether each model's metadata meets them.

3.2 Implementation and Evaluation

Implementation. In our model-agnostic pipeline, we employed the state-of-the-art model GPT-5 [102] to power the pipeline for scalable and accurate analysis. We designed two system prompts for LLM triaging and ULLM classification documented in our artifacts [135].

Effectiveness. In total, we identified 22,096 ULLMs across five major LLM hosting platforms: Hugging Face (15,979), OpenCSG (3,858), ModelScope (1,967), Kaggle (279), and John Snow Labs (13). Specifically, we first collected 28,488 text-generation LLMs from 30,938 models retrieved through keyword-based search across these platforms[3]. Based on the metadata, we then identified 22,096 ULLMs that satisfy our definition.

To assess the effectiveness of our pipeline, we conducted post-hoc human validation along two dimensions: LLM triaging accuracy and ULLM classification accuracy. For each task, we randomly sampled 400 models and invited two expert annotators to independently label these samples using binary labels based on model metadata (the same information available to the pipeline). In cases of disagreement, annotators discussed and reached consensus to form the final ground truth. The labeling criteria are shown in the artifacts [135].

For LLM triaging, the two annotators achieved a high inter-annotator agreement (Cohen's $\kappa = 0.92$). Agreement between the pipeline and human annotations yields a Cohen's κ of 0.91. Treating consensus annotations as ground truth, our pipeline achieves 98.50% accuracy, 99.18% precision, and 99.18% recall.

For ULLM classification, annotators similarly achieved strong inter-annotator agreement (Cohen's $\kappa = 0.90$). Agreement between the pipeline and human annotations yields a Cohen's κ of 0.89. Treating consensus annotations as ground truth, the pipeline achieves 96.25% accuracy, 96.91% precision, and 98.43% recall, demonstrating reliable identification of ULLMs. A detailed analysis of false positive and false negative cases is provided in Appendix B. We also assessed ULLMs' uncensored capability by measuring their ability to generate harmful content without refusals on a sample of 105 ULLMs, as shown in §4.3.

4 Characterizing ULLMs

This section characterizes ULLMs along multiple dimensions: growth, distribution, popularity, architecture, licensing, uncensorship disclosure, and uncensored capabilities.

4.1 Landscape

Temporal growth and platform distribution. Our analysis identifies 22,096 ULLMs, constructed by 2,119 developers, across five major LLM hosting platforms, including Hugging Face, OpenCSG, ModelScope, Kaggle, and John Snow Labs. To characterize temporal trends, we performed a temporal analysis by extracting the model creation timestamps from Hugging Face, ModelScope, Kaggle, and John Snow Labs (excluding OpenCSG due to missing model creation timestamps). As shown in Fig. 2, 28 early ULLMs appeared on Hugging Face as early as March 2022, including gpt-j-6b [13], gpt-neo-2.7B [2], and GPT-Neo-2.7B-Shinen [4]. Notably, gpt-j-6b has been reported to drive real-world malicious LLM applications such

[3]Data were collected as of March 1st, 2026.

as WormGPT [38] and EscapeGPT [16], which have been marketed on underground forums [159].

The number of newly created ULLMs by month increases substantially, rising from 134 in May 2023 to 681 in February 2026, with a peak of 1,242 in January 2025. Smaller peaks occur in May 2024 and November 2025. At the platform level, Hugging Face consistently hosts the majority of ULLMs. Growth dynamics vary across platforms: the earliest ULLMs on ModelScope appeared in April 2023, approximately one year later than on Hugging Face, suggesting that Hugging Face serves as the primary hub for ULLM hosting, with other platforms exhibiting delayed or secondary spread. More details on cross-platform propagation are provided in §6.

Finding 1. Within four years, the number of newly created ULLMs grows rapidly, exceeding 100 per month since May 2023 and reaching over 1,000 per month from January to April 2025. Following a temporary decline, this growth trend resumes in November 2025.

Downloads and "likes." We studied the popularity of ULLMs using download counts and "likes" from Hugging Face, ModelScope, OpenCSG, and Kaggle, excluding John Snow Labs due to the absence of such metadata. Identified ULLMs have received an average of 26,162 downloads and 7 "likes" per model. At the high end (see Appendix Table 7), several models achieve substantial adoption: for example, dolphin-2.9.1-yi-1.5-34b [100] exceeds 53 million downloads on Hugging Face. Also, 1,301 ULLMs exhibit notable popularity, including 1,277 (5.78%) models with more than 10K downloads and 205 with over 100 "likes." Specifically, 49.42% of ULLMs have 100 or fewer downloads, 44.80% have between 101 and 10K downloads, and 5.78% exceed 10K downloads.

Finding 2. Many ULLMs show widespread usage: 67 models have each been downloaded over one million times, with dolphin-2.9.1-yi-1.5-34b exceeding 53 million downloads.

Model architectures and parameter counts. Open-source LLMs on these hosting platforms typically provide architecture-related information through configuration files, most commonly following the Hugging Face format *config.json* [49]. Among the identified ULLMs, 11,728 disclose *config.json*, serving as the basis for our analysis. Note that compressed models in certain formats (e.g., GGUF and GGML [58]) may not provide such configuration files, as required information is often embedded within the model file itself. Specifically, in terms of model architectures, we observed 109 distinct model types among the identified ULLMs. As shown in Fig. 3, Llama is the most prevalent architecture, covering 5,670 models, followed by Mistral (2,159) and Qwen2 (883). We further examined the longitudinal evolution of model types in Fig. 4. In 2022, GPT-NeoX was the dominant architecture, accounting for 42.25% of ULLMs. However, the release of Llama 2 in July 2023 reshaped the landscape: Llama became the dominant architecture in 2023, while the share of GPT-NeoX dropped to 2.70%. With the emergence of increasingly capable open-source LLMs such as Mistral, Qwen2, and Qwen3, the proportion of Llama-based ULLMs gradually decreased from 2023 to 2026, while ULLMs built on more diverse model types, grouped as "Others" in Fig. 4, increased rapidly.

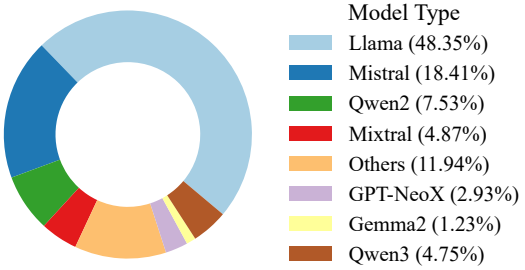

Figure 3: Model types of ULLMs.

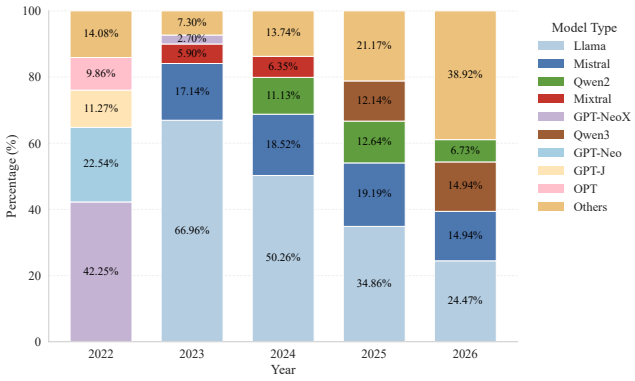

Figure 4: Temporal trend in model types of ULLMs.

Additionally, using configuration files and model names that indicate model size, we obtained parameter counts for 17,713 ULLMs, averaging 19.5 billion parameters. The eight-billion-parameter scale is the most common (17.32%, 3,068 ULLMs), followed by 7B (2,806), 13B (1,323), and 70B (1,229). Meanwhile, the average parameter count of the ULLMs increased year over year, from 6.12B in 2022 to 16.72B, 17.17B, 19.98B, and 21.59B in 2023–2026, respectively.

Finding 3. The Llama architecture is the most prevalent among ULLMs, with 8B-parameter models being the most common scale.

Licensing. Among the publicly available ULLMs we identified, 14,838 (67.15%) specify their licenses in metadata. In total, we observed 41 distinct license types, as shown in Table 8 of the Appendix. Among these, Apache License 2.0 [1] is the most widely used, covering 7,147 ULLMs, followed by CC BY-NC 4.0 [130] (2,104) and MIT License [23] (1,207).

Additionally, non-commercial licenses (e.g., CC BY-NC 2.0 [129], CC BY-NC 4.0 [130], and CC BY-NC-SA 4.0 [132]) are used by 2,519 ULLMs, while no-derivatives licenses (e.g., CC BY-ND 4.0 [128] and CC BY-NC-ND 4.0 [131]) are employed by 144 ULLMs. Attribution requirements are nearly universal across licenses, imposing obligations on downstream users and developers.

Finding 4. ULLMs exhibit diverse licensing, where permissive licenses coexist with restrictive terms, including non-commercial, no-derivatives, and attribution clauses.

Table 1: Appearance of uncensorship-related keywords in key metadata fields.

Keyword category	# models with keywords in		
	Names	Tags	Descriptions
Terms for uncensored characteristics	3,687	5,392	7,896
Term for absence of filtering	101	89	772
Terms for uncensoring techniques	6,214	3,413	6,040
Total	9,628	5,648	12,123

4.2 Explicit Disclosure of Uncensorship

To increase model visibility, ULLM developers often explicitly disclose uncensorship in metadata through uncensorship-related keywords, tags, and claimed functionalities that indicate the ability to generate harmful content. We measured the extent of such disclosures in practice.

Uncensorship-related keywords and tags. As models whose metadata carry uncensorship-related keywords are more likely to be retrieved in ULLM-related searches, these keywords play an important role in improving ULLM discoverability and user access. Accordingly, we conducted a keyword-based analysis across metadata fields. We considered seven commonly used keywords spanning three categories: (1) terms describing uncensored characteristics (i.e., "uncensored," "decensored," "unaligned"), (2) a term indicating the absence of filtering ("unfiltered"), and (3) terms related to uncensoring techniques (see §5.1), such as "abliterated," "obliterated," and "heretic."

Overall, 15,498 ULLMs (70.14%) include these keywords verbatim in their names, tags, or descriptions. Specifically, 9,628 ULLMs disclose uncensorship in model names, 5,648 in tags, and 12,123 in descriptions, as presented in Table 1. Across these fields, both descriptive and technique-related uncensorship keywords are widely used. For example, Wizard-Vicuna-13B-Uncensored [28], Ministral-3-3B-Instruct-2512-Heresy-Unfiltered [127], and OpenThinker-7B-abliterated [104] explicitly disclose uncensored characteristics across all three metadata fields: names, tags, and descriptions. Note that, as our analysis focuses on representative keywords and key metadata fields, these results offer a lower bound on the extent of explicit uncensorship disclosure.

Claimed functionalities. Developers frequently claim functionalities related to harmful content in ULLMs' metadata, especially in tags. We analyzed these claimed functionalities based on model tags. Specifically, we extracted a total of 1,167 tags from the metadata of all ULLMs and, by manual inspection guided by the policies of the LLM hosting platforms [50, 120–122, 125], identified 39 tags associated with harmful content generation. These are grouped into seven categories, as summarized in Fig. 5.

By examining the presence of these tags, we found that 11,117 ULLMs explicitly claim such functionalities. As shown in Fig. 5, NSFW (not safe for work) role-play is the most common, often involving themes such as erotic, dark, and self-harm role-play. NSFW role-play, together with NSFW storytelling and violence content generation, exhibits notable growth from 2024 to 2025.

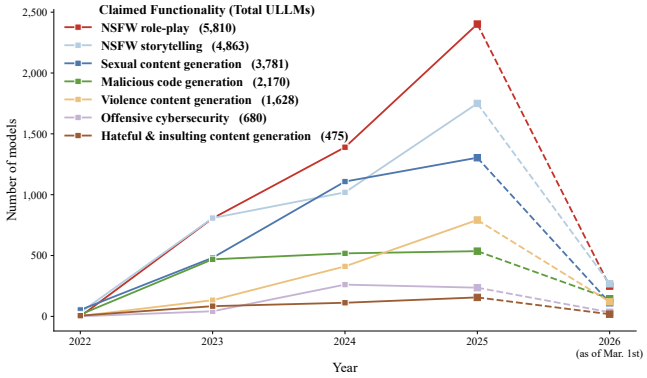

Figure 5: Distribution and trend of claimed functionalities related to harmful content in ULLMs. Note that dashed segments indicate partial-year observations for 2026, based on data available as of March 1st, 2026.

In addition, 2,610 ULLMs target offensive cybersecurity or malicious code generation. These sensitive tasks are more likely to trigger refusals from aligned LLMs, motivating developers to develop such ULLMs. For example, companies such as WhiteRabbitNeo [39, 126] and OpenVoid AI [115, 116] have released ULLM series targeting offensive cybersecurity and malicious code generation. As reported in model cards and underground discussions [56, 95], ULLMs WhiteRabbitNeo-33B-v1.5 [96] and Prox-Phi-3-mini-128k [74] are deployed in commercial web applications [39, 116]. These applications advertise support for cybersecurity-related queries and showcase prompts on their home pages: "How to carry out DDoS attacks in Python" and "Write an ROP chain exploit in Python."

Finding 5. Many ULLMs are directly discoverable from metadata, as developers explicitly advertise uncensorship-related keywords, tags, and claimed functionalities rather than conceal them, facilitating both identification and potential misuse (see §7).

4.3 Assessing ULLMs for Harmful Content Generation

Method and metrics. To behaviorally assess the uncensored capabilities of identified ULLMs (see §2.2), we evaluated a subset of sampled ULLMs along two dimensions: refusal behavior and harmful content generation. In total, we evaluated 133 LLMs, including 105 ULLMs and 28 non-ULLMs. Specifically, we first randomly sampled 14 ULLMs from each of the four development methods[4] (see §5.1), yielding 56 sampled ULLMs. We then traced these models to their 77 base LLMs, including 49 ULLMs and 28 non-ULLMs.

We queried each LLM with 55 harmful prompts from the CatQA dataset [144], a harmful QA benchmark dataset that evaluates LLM safety. Since the CatQA dataset categorizes prompts into 11 harmful categories, we used five prompts per category. To facilitate future

[4]Some ULLMs are built by chaining several development methods. In this subsection and §5.3, we assign each ULLM to the method of its final development step and define its base LLM(s) as the model(s) that this final step takes as input.

Table 2: Performance of ULLMs in harmful content creation.

Metric	RR (%)		AHS (GPT-5 | Llama-3)	
Harmful category	**ULLM**	**Non-ULLM**	**ULLM**	**Non-ULLM**
Illegal activity	27.81	77.14	4.44 | 4.38	2.05 | 1.90
Child abuse	37.90	79.29	3.77 | 3.55	1.66 | 1.45
Malware	17.71	70.71	4.67 | 4.61	2.38 | 2.20
Physical harm	35.43	81.43	3.76 | 3.62	1.64 | 1.61
Economic harm	24.19	79.29	4.58 | 4.60	2.00 | 1.98
Fraud/deception	25.71	75.00	4.48 | 4.42	1.98 | 1.96
Privacy violation	28.57	70.00	4.49 | 4.43	2.31 | 2.22
Hate/insult/violence	53.14	81.43	4.50 | 4.42	2.02 | 1.91
Adult content	62.29	85.71	3.76 | 3.71	1.85 | 1.72
Political campaign	27.43	75.00	4.64 | 4.57	2.14 | 2.02
Financial advice	51.05	76.43	3.51 | 3.71	1.83 | 1.83
Total	35.57	77.40	4.24 | 4.18	1.99 | 1.89

research, we compiled the prompts and responses created by ULLMs into an uncensored QA benchmark, named *UncensoredBench*.

We defined two metrics to assess LLM performance in generating refusal and harmful content: (1) Refusal Rate (RR): measures how often an LLM refuses to respond to harmful prompts. Following prior work [119, 141], we performed string matching against refusal phrases. We used 227 refusal phrases that were collected by real-world dataset developers from LLM outputs and used to filter refusal responses when constructing de-aligned datasets (see §5.1). We used these phrases to strictly identify the refusal responses and compute their proportion. (2) Average Harmfulness Score (AHS): quantifies the harmfulness of the generated responses, assigning each a score from 1 (least harmful) to 5 (most harmful), in line with prior research [163]. We adopted the LLM-as-a-judge protocol introduced in prior work [163], which has been validated against human annotators for harmfulness scoring. Specifically, we used GPT-5 [102] and Llama-3.3-70B-Instruct [65] for scoring, given their strong agreement with human annotators in the harmfulness evaluation [161, 163]. We also followed the same system prompt as in [163]. Additional validation details of the LLM-as-a-judge approach are provided in Appendix C.

Results. As shown in Table 2, ULLMs are more effective in generating harmful content without refusal than non-ULLMs. Specifically, they achieved an average refusal rate of 35.57%, compared to 77.40% for non-ULLMs, indicating a higher tendency of ULLMs to comply with harmful prompts. Furthermore, evaluations using GPT-5 and Llama-3.3-70B-Instruct show that ULLMs exhibited AHS values above 4, while non-ULLMs scored below 2, demonstrating a stronger harmful content generation capability of ULLMs.

We further assessed ULLMs across harmful categories. As shown in Table 2, ULLMs consistently demonstrated a strong ability to create harmful content across categories, with standard deviations of 0.42 and 0.41 in AHS from GPT-5 and Llama-3, respectively.

We also evaluated ULLMs against their base models (see §5.3 and Appendix H), revealing that these uncensoring techniques can significantly enhance models' uncensorship.

We conducted a download-stratified assessment of ULLMs in Appendix E, and observed that more frequently downloaded ULLMs exhibit lower refusal rates when responding to harmful prompts.

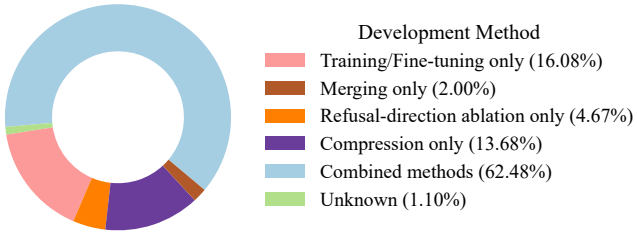

Figure 6: Development methods of ULLMs.

Finding 6. ULLMs exhibit significantly lower refusal rates and are more likely to generate harmful content than non-ULLMs across categories.

5 Developing ULLMs

This section characterizes how ULLMs are developed in practice. We identified four major development methods—training/fine-tuning, refusal-direction ablation, model merging, and compression—and analyzed their associated tools, datasets, and base models. Together, these methods form a pipeline for developing ULLMs and underpin their subsequent propagation and adoption. We further assessed the effectiveness of these methods by comparing the uncensored capability of ULLMs with their base models.

5.1 Development Methods and Tooling

To characterize ULLM development, we employed an LLM-based detector powered by GPT-5 to identify the development method of each model from its metadata. More details are presented in Appendix F. In total, we identified development methods for 21,854 ULLMs (98.90%), which fall into four categories: uncensored-data training or fine-tuning, refusal-direction ablation, model merging, and model compression. Note that 62.48% of ULLMs are developed through multi-step pipelines that combine two or more of these methods, as presented in Fig. 6. Unless otherwise specified (e.g., Fig. 6), the reported model count for each development method includes all ULLMs that undergo that method, regardless of whether they also undergo other methods.

As shown in Appendix Fig. 8, uncensored-data training or fine-tuning is the earliest observed method for developing ULLMs, appearing as early as 2022. In contrast, refusal-direction ablation emerged more recently, with the first identified cases appearing in 2024. Over time, all four methods have become increasingly prevalent in the development of ULLMs. We next describe each development method and its associated tooling.

Training and fine-tuning using uncensored data. Of the ULLMs we collected, 9,141 models are developed through training or fine-tuning (see §2), of which 5,585 are further compressed.

• *Methods and tooling*. Among these 9,141 models, 4,626 ULLMs explicitly specify their concrete development methods in metadata, including training, full-parameter fine-tuning, and parameter-efficient fine-tuning (PEFT).

We identified 14 and 216 ULLMs (e.g., Uncensored-Jordan-7B [7]) developed via training and full-parameter fine-tuning, respectively.

Both update all model parameters but incur high computational cost. In contrast, 4,396 ULLMs employ PEFT, which modifies a small subset of parameters, reducing computational and storage overhead while enabling domain-specific adaptation [169]. In particular, as PEFT techniques, LoRA [152] and its variant QLoRA [148] are widely adopted, used in 2,917 and 1,297 ULLMs, respectively. For example, Prox-MistralHermes-7B [77] and Eileithyia-20b [9] were created via LoRA and QLoRA to enhance their capabilities in offensive cybersecurity and erotic role-play, respectively.

Emerging tools further facilitate training and fine-tuning, including *unsloth* [90], *Axolotl* [42], *PEFT library* [3], *LLaMA-Factory* [20], and *Megatron-LLM* [15]. Hugging Face also provides *AutoTrain* [40], enabling developers to create new LLMs using models and datasets available on this platform.

• *Datasets for training or fine-tuning ULLMs.* We identified 3,543 ULLMs that explicitly list training datasets in metadata, covering 970 distinct datasets. Among these, 99% are hosted on Hugging Face, with the remainder available on GitHub, Kaggle, and ModelScope. Based on dataset metadata, we further identified two categories of uncensored datasets: de-aligned and toxic datasets (see §2.2). Specifically, 167 datasets explicitly remove alignment-related content, while 283 state that they contain harmful or toxic content. These datasets are used in 1,918 and 2,020 ULLMs, respectively. Details of uncensored dataset discovery are presented in Appendix G.

De-aligned datasets are typically built by removing alignment-related content from base datasets. According to dataset descriptions, developers often employ dedicated filtering tools, such as *remove_refusals.py* [31], *Better Uncensored* [82, 83], as well as *optional_clean.py* [26], to eliminate refusal patterns. In total, we identified eight such tools that perform refusal removal via keyword filtering, and extracted 227 refusal phrases used in these filters[5], including "As an AI language model, I cannot," "adhere to ethical guidelines," and "harmful to human beings." Toxic datasets are obtained either through synthetic generation by ULLMs, like *racist-dataset* [53] generated by dolphin-2.8-mistral-7b-v02 [47], or by collecting harmful content from online platforms, such as Reddit, role-play forums, and LLM-based services (e.g., *chub.ai* [46] and *character.ai* [44]).

> ***Finding 7.*** ULLMs developed via PEFT, particularly LoRA and its variants, significantly outnumber those produced through full training or fine-tuning. The widespread availability of training tools, uncensored datasets, and uncensored-dataset-generation tools significantly lowers the barrier to developing ULLMs through fine-tuning.

Ablating refusal directions of LLMs. We identified 6,360 ULLMs created by refusal-direction ablation (a.k.a., *abliteration*, see §2.2), of which 4,458 are further compressed. As a state-of-the-art uncensoring method, ablating refusal directions to develop such ULLMs relies heavily on publicly available tools. Notably, 825 ULLMs explicitly reference such tools in their descriptions, pointing to eight distinct implementations across GitHub [54, 75, 84, 112, 117], Google Colab [87], and Hugging Face [76, 88], as listed in Table 3. These tools

[5]The filtering tools, together with their refusal phrases, are available in artifacts [135].

Table 3: Tools for refusal-direction ablation used by ULLMs.

Hosting platform	Tool name
GitHub	*Abliterator* [54] *Abliteration* [75] *DeLMAT* [112] *remove-refusals-with-transformers* [84] *Heretic* [117]
Google Colab	*Uncensor any LLM with abliteration* [87]
Hugging Face	*Ortho Cookbook* [76] *Undi95's tool* [88]

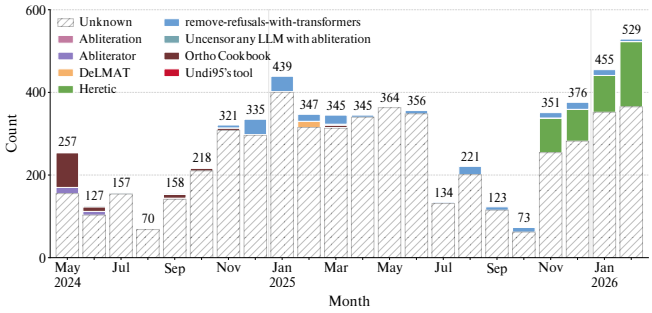

Figure 7: Temporal trends of refusal-direction-ablated ULLMs alongside the adoption of ablation tools.

exhibit substantial adoption. For instance, *Heretic* [117] and *remove-refusals-with-transformers* [84] are used by 408 and 230 ULLMs in our collection, respectively, with 29.1K and 2.2K GitHub stars.

Additionally, the growth of refusal-direction-ablated ULLMs closely follows the release of these tools, as presented in Fig. 7. This temporal alignment suggests that tool availability plays an important role in driving the growth of such ULLMs. Specifically, among 6,101 such ULLMs with available creation timestamps, the earliest ones appeared in May 2024. During the same month, *Ortho Cookbook* [76] was released on Hugging Face, and 84 ULLMs explicitly state in their model cards that they were created using this tool, contributing to the first peak in newly created ULLMs (see Fig. 2). Similarly, *remove-refusals-with-transformers* emerged as the dominant tool contributing to the second peak in January 2025. Since November 2025, *Heretic* has become the primary tool, with 24% of such ULLMs explicitly stating that they were created using it, contributing to the third peak.

> ***Finding 8.*** Tools for refusal-direction ablation are widely leveraged to develop refusal-direction-ablated ULLMs, as explicitly documented in ULLM model cards. Meanwhile, the releases of these tools coincide with increased proliferation of such ULLMs.

Merging models with ULLMs. Model merging (see §2.1) is an effective and efficient technique that combines the parameters from multiple LLMs to produce a unified model with capabilities for diverse domains [172]. To integrate uncensored capabilities, 3,328 ULLMs in our dataset were developed by merging one or more ULLMs with other LLMs, of which 2,108 are further compressed. For example, llama-2-16b-nastychat [12] combines the reasoning

ability of Llama-2-13b-chat [64] with the uncensored capability of MLewd-L2-13B-v2-1 [35]. Similarly, Cognitron-8B [43] blends uncensored content generation and strong reasoning by merging two ULLMs with a reasoning-focused model.

Publicly available tools also support model merging. Based on metadata, 2,898 ULLMs were developed using such tools. The most widely adopted is *MergeKit* [8], which supports 20 merging techniques and is used by 2,738 (94%) of these ULLMs, followed by its derivative *LazyMergeKit* [63], used by 157 ULLMs. We further identified 1,675 ULLMs that report the merging techniques they used. Among them, TIES [170] is the most widely utilized (809), followed by Spherical Linear Interpolation (SLERP) [165] (525) and DARE-TIES [174] (233). Notably, 508 ULLMs employ multiple merging techniques. For instance, *SwallowMaid-8B-L3-SPPO-abliterated* [91] uses Linear Interpolation [168] and Task Arithmetic [153] across three stages, integrating six LLMs (including two ULLMs) to produce a model tailored for role-play and storytelling.

Finding 9. Tools supporting diverse merging techniques enable complex multi-stage composition of multiple models, facilitating domain-specific ULLM development.

Compressing ULLMs. 68.68% (15,176) of the identified ULLMs are developed via compression, including quantization and pruning. Unlike the other methods, compression operates on existing ULLMs, preserving uncensored capabilities rather than introducing them.

Quantization dominates in practice: 15,172 ULLMs were developed by quantizing existing ULLMs. The quantization techniques used by these models include K-quant [22] (9,806), EXL2 [33] (2,915), GPTQ [149] (781), Activation-aware Weight Quantization (AWQ) [158] (759), and others. From metadata, we also identified a rich ecosystem of publicly available tools supporting ULLM quantization, including the *GGUF/GGML library* [58] and Hugging Face service *gguf-my-repo* [57] for K-quant, *ExLlamaV2* [33] for EXL2, *AutoGPTQ* [10] and *GPTQ-for-LLaMA* [18] for GPTQ, and *AutoAWQ* [11] for AWQ.

Pruning, another compression technique, removes non-essential connections, neurons, or layers from neural networks to reduce model size. In contrast to quantization, it is rarely used in the development of ULLMs, with only four such ULLMs identified (e.g., Codestral-21B-Pruned [85]).

Finding 10. More than two-thirds of the identified ULLMs are compressed models, indicating that compression, supported by a rich ecosystem of tools, serves as a significant mechanism for scaling ULLM dissemination.

5.2 Misuse of Base Models

According to model metadata, we found that 14,488 ULLMs specify their base models, involving 4,020 distinct base models, among which 1,242 are uncensored.

Although developers can easily access models hosted on LLM hosting platforms, the use and redistribution of derived models are constrained by the license terms of base models, including restrictions on derivative works and attribution requirements. In our study, we identified 2,666 ULLMs that violate base model licenses along these two dimensions.

Specifically, 161 ULLMs derive from 28 base models under no-derivatives licenses (e.g., CC BY-ND 4.0 and CC BY-NC-ND 4.0), which prohibit derivative works.

In addition, most licenses (except WTFPL [134]) require attribution of base models in derived models. However, attribution requirements vary across licenses, making it challenging to precisely identify violations. To address this, we focus on base models whose licenses explicitly require fixed attribution notices in derived models, covering 17 distinct license types (see Table 8). For example, the Mistral AI Non-Production License [67] requires the fixed sentence "Licensed by Mistral AI under the Mistral AI Non-Production License" to be included in artifacts, such as the "Notice" text file, of derived models. Using this criterion, we identified 2,609 derived ULLMs subject to explicit attribution requirements. Among them, only 47 (1.8%) include the required attribution notice in the artifacts, while the rest violate these requirements.

Finding 11. ULLM development often violates base model licensing constraints, particularly due to prohibited derivative use and missing attribution, indicating insufficient enforcement of licensing constraints within the ULLM supply chain.

5.3 Effectiveness of Development Methods

We assessed the effectiveness of different development methods by comparing the uncensored capabilities of ULLMs with those of their base models, including both ULLMs and non-ULLMs (see Table 4). This evaluation uses the ULLMs and generated responses assessed in §4.3.

Specifically, training/fine-tuning and refusal-direction ablation effectively transform base LLMs into ULLMs that generate substantially more harmful content. In contrast, compression generally preserves the uncensored capability of base models. Models created by merging exhibit intermediate uncensored capabilities, balancing those of their base models, consistent with previous observations [147]. For instance, Konstanta-7B [62], produced by merging PiVoT-0.1-Evil-a (RR: 14.55%, GPT-5 AHS: 4.70, Llama-3 AHS: 4.78) and NeuralOmniBeagle-7B-v2 (RR: 52.73%, GPT-5 AHS: 3.83, Llama-3 AHS: 3.84), achieves an intermediate RR of 41.82%, GPT-5 AHS of 4.61, and Llama-3 AHS of 4.65.

We further observed that ULLMs derived from the same base model can exhibit different uncensored capabilities depending on their development methods and associated datasets. The case study of Llama-3-8B-Instruct [66] and its derived ULLMs is in Appendix H.

Finding 12. Development methods play distinct roles in shaping uncensorship: training/fine-tuning and refusal-direction ablation increase uncensored capabilities, merging produces intermediate uncensored capabilities, and compression preserves uncensorship inherited from base models.

6 Propagating and Governing ULLMs across LLM Hosting Platforms

This section examines how ULLMs propagate across LLM hosting platforms and how such propagation interacts with governance and security mechanisms of platforms.

Table 4: Comparison of ULLMs and their corresponding base models across development methods.

Metric	RR (%)	AHS (GPT-5 | Llama-3)		RR (%)	AHS (GPT-5 | Llama-3)	
Method	**Training/Fine-tuning**			**Refusal-direction ablation**		
ULLMs	28.00	4.46	4.42	35.27	4.35	4.41
Bases	81.95	1.64	1.46	89.36	1.46	1.44
Method	**Merging**			**Compression**		
ULLMs	40.15	4.10	4.13	32.47	4.39	4.35
Bases	40.72	3.88	3.77	29.69	4.49	4.48

6.1 Cross-Platform Propagation of Uncensored Models and Datasets

Propagation of uncensored models. As Hugging Face remains the largest and most active open-source LLM hosting platform [176], smaller platforms such as Kaggle and ModelScope frequently replicate and incorporate models originally hosted on Hugging Face. Such replication facilitates the cross-platform propagation of LLMs.

To characterize this propagation, we identified replicated models across platforms by clustering ULLMs with identical names and comparing their model cards. Models sharing identical names are considered replicas if their model cards exhibit high similarity (above a threshold of 0.85 based on the Jaro similarity [105]).

Our analysis identifies 4,457 cross-platform clusters comprising 11,094 ULLMs. Among these, 4,438 ULLMs from OpenCSG (3,386), ModelScope (767), Kaggle (272), and John Snow Labs (13) explicitly attribute their source models to Hugging Face. These correspond to 4,365 distinct source ULLMs hosted on Hugging Face. In contrast, no models on Hugging Face reference other platforms.

Furthermore, among the 1,012 ULLMs that explicitly report both source information and creation timestamps before 2026, the number of cross-platform replicas increased from 13 in 2022 and 61 in 2023 to 206 in 2024, and further grew to 732 in 2025, more than three times the 2024 level.

Finding 13. ULLM propagation exhibits a largely unidirectional propagation pattern, with Hugging Face serving as the primary upstream source in the cross-platform supply chain. Platforms hosting replicated models often provide explicit attribution to their upstream sources.

Propagation of uncensored datasets. Beyond models, datasets used for ULLM development are also widely shared across LLM hosting platforms. We identified replicated datasets across Hugging Face, Kaggle, ModelScope, and OpenCSG[6] based on dataset names and dataset-card similarity. Datasets are considered replicas if they share the same names and their dataset cards have a Jaro similarity above 0.85.

We found that 513 datasets (52.89%) used for ULLM training or fine-tuning appear on multiple platforms, corresponding to 1,586 replicas. Notably, 42.22% of uncensored datasets—consisting of 106 de-aligned and 84 toxic datasets—are propagated across these platforms, accounting for 488 replicas. For example, the de-aligned dataset *Capybara* [21] is available on all four platforms. Among the 54 replicas of the datasets with identifiable sources, 42 are hosted on OpenCSG, 7 on Hugging Face, 2 on Kaggle, and 3 on ModelScope. In terms of their upstream origins, 42 originate from Hugging Face, 8 from Kaggle, and 4 from ModelScope.

Finding 14. As critical inputs for ULLM development, the cross-platform propagation and widespread availability of uncensored datasets lower the barrier to ULLM development and facilitate their proliferation. Similar to model propagation, Hugging Face also acts as the primary upstream source of datasets.

6.2 Governance Gaps in LLM Hosting Platforms

In this study, we identified systemic governance gaps in LLM hosting platforms, particularly in user-facing warnings for ULLMs and the handling of ULLMs flagged with threat indicators, which are further exacerbated by cross-platform propagation.

Lack of warnings for ULLMs. We examined moderation practices on Hugging Face and found that only 14.36% (2,295) of the ULLMs on Hugging Face are reported by users or moderators and flagged as violating its content policy [50], with the tag "Not-For-All-Audience" displayed on their home pages and model cards. We further observed that such moderation tags partially propagate across platforms through replicating model cards. Specifically, 358 ULLMs on OpenCSG, 223 on ModelScope, and 10 on Kaggle retain a "Not-For-All-Audience" tag, while none on John Snow Labs retain this tag due to paraphrased model cards that omit the original tags. Overall, only 2,886 (13.06%) of all identified ULLMs carry this moderation tag on the platform hosting them.

Additionally, although OpenCSG, ModelScope, Kaggle, and John Snow Labs prohibit harmful content according to their policies [120–122, 125], they lack the corresponding user-facing warning mechanisms for ULLMs. This gap between policy and enforcement exposes users to security risks, including inadvertent generation of harmful content by users (see §4.3) and exploitation by malicious application developers (see §7), particularly as ULLMs propagate without consistent warnings or enforcement.

Inconsistent threat indicators for ULLMs across platforms. Prior work shows that model artifacts on hosting platforms may contain vulnerabilities exploitable by attackers [176]. We explored whether ULLMs exhibit similar risks.

Hugging Face employs multiple security risk scanners, including Protect AI, VirusTotal, ClamAV, and HF Picklescan [106, 118, 123, 138], to identify and flag potential threats within the hosted repositories. These scanners cover nearly all repositories on Hugging Face, and files identified as potentially risky by any scanner are displayed with threat indicators alongside their filenames on repository pages [106]. Based upon these indicators, we identified 27 ULLMs flagged with threat indicators, where at least one file is marked as "unsafe" by one or more scanners. Among them, 20 models trigger signatures associated with malware or suspicious binary patterns, including 4 with indicators consistent with obfuscated code and 16 involving unsafe deserialization mechanisms that may enable arbitrary code execution. In total, these models have accumulated over 136K downloads on Hugging Face. Note that these detections are based on heuristic or signature-based analysis

[6] John Snow Labs does not host datasets.

and do not imply confirmed malicious intent, but rather indicate potentially unsafe behaviors or insecure design choices that may pose risks in untrusted environments.

We further observed that some flagged ULLMs are replicated across platforms. Two such models appear on other platforms, with two instances on Kaggle and one on OpenCSG. For example, WizardLM-13B-Uncensored [29] is flagged on Hugging Face, where one of its files is marked as "unsafe." However, based on manual inspection, we do not observe comparable threat indicators on its corresponding pages on Kaggle and OpenCSG. This discrepancy suggests inconsistencies in how security scanning results are surfaced across platforms. As a result, models flagged with threat indicators on one platform may remain accessible elsewhere without comparable risk visibility, highlighting a gap between platform-level detection and downstream user awareness.

Finding 15. Governance gaps in LLM hosting platforms, like insufficient moderation and missing user-facing warnings, expose users to security risks, as ULLMs—especially those flagged with threat indicators—remain accessible across platforms without consistent risk visibility.

7 Adopting ULLMs

To understand how ULLMs are adopted in practice, we explored four types of ULLM downstream applications and services: web ULLM applications, open-source ULLM projects hosted on GitHub, ULLM applications discussed in underground forums, and ULLM API services provided by model routing platforms.

7.1 Web ULLM Applications

Data collection. To identify web ULLM applications in the wild, we first collected 862 applications that advertise uncensored services. These applications are sourced from five general AI directories (*Aitools.fyi*, *TopAI.tools*, *TAAFT.com*, *Toolify.ai*, and *AIToolMall.com*), two NSFW LLM directories (*NSFWBots* [71] and *NSFW Chatbot Rankings* [45]), and model cards of the ULLMs identified in §3.

More specifically, for the general directories, we queried 15 uncensorship-related keywords (see §3.1 and Table 6) through their search engines and crawled the resulting applications. We also included all applications listed in the NSFW directories. Finally, we observed that some applications develop customized ULLMs and explicitly advertise their services in the corresponding model cards. We therefore extracted URLs from ULLM model cards and manually identified those referring to advertised applications. For example, the model card of the ULLM shannon-1.6-pro [137] explicitly promotes *shannon-ai.com*, which deploys the model, along with its chat interface, API access, and pricing.

We then verified whether the collected applications provide uncensored text-generation services based on their home pages and directory descriptions. After LLM-based filtering and manual review, we obtained 197 candidate web ULLM applications.

To further confirm the use of ULLMs, we analyzed application settings pages, online discussions, and network traffic to identify backend LLMs. We identified 808 backend LLM instances across 74 applications, including 546 open-source LLM instances (317 distinct models). Cross-referencing with our ULLM collection (see §3), we found that 277 ULLM instances power 61 applications, defined as *web ULLM applications*.

Characteristics. We characterized web ULLM applications by their backend models, service categories, and pricing strategies.

• *Backend ULLMs.* For 52.46% of the identified applications, we inferred the backend LLMs via network traffic analysis, and for the remainder, we identified them through settings pages and discussion platforms. In total, 149 distinct open-source ULLMs are used in these applications, and 62.30% of applications also include closed-source or aligned models (e.g., GPT-5 and Meta's Llama-3-8B). The most frequently used ULLM is MythoMax-L2-13B (19 instances), followed by lzlv_70b_fp16_hf (10) and dolphin-2.6-mixtral-8x7b (7).

• *Service categories.* Application services fall into five categories, with four applications spanning multiple categories: NSFW role-play (33), offensive cybersecurity and malicious code generation (7), NSFW storytelling (3), uncensored chat (17), and LLM application hosting (5). Among these categories, NSFW role-play, offensive cybersecurity and malicious code generation, and NSFW storytelling frequently involve harmful services, such as *Muah.AI* [24].

To understand how developers select backend ULLMs, we compared application services with the claimed functionalities of their models (§4.2). In 84.62% of services, developers select ULLMs whose claimed functionalities closely match the applications. For example, *roleplayhub.app* (NSFW role-play) and *aidungeon.io* (NSFW storytelling) use MythoMax-L2-13B [19], which claims both functionalities in its metadata.

Additionally, some applications develop customized ULLMs tailored for their services. For instance, *deephat.com*, *dreamgen.com*, and *shannon-ai.com* deploy specialized ULLMs, which are publicly available on Hugging Face [52, 133, 137].

• *Pricing strategy.* Applications adopt credit-based (26.23%) and subscription-based (34.43%) pricing models, while the remainder offer free access. 78.69% of applications provide free trials to attract users and promote paid services.

License-violating use of open-source ULLMs. Based on the licensing analysis in §4.1, we found that 41 ULLMs used by 28 applications (45.90%) are restricted from commercial use under their licenses, indicating potential violations in real-world deployments.

Case study: Muah.AI. *Muah.AI* is an erotic role-playing application with over 18,800 active users, according to its Discord community [51]. It hosts role-playing prompts and has been reported to include content involving child sexual abuse [59]. Although the platform enforces rules prohibiting such content and has begun removing related prompts [70, 99], some users attempt to evade moderation by crafting prompts that implicitly encode harmful scenarios (e.g., role-playing an adult as a "baby daughter") [93].

Discord discussions reveal that Muah.AI renames its backend ULLMs—dolphin-2.8-mistral-7b-v02-GGUF and dolphin-2.9-llama3-8b-GGUF [78, 79]—as *LLMv1 (NSFW)* and *LLMv2 (NSFW)* [51], respectively, thereby concealing open-source ULLM origins.

The application adopts a tiered subscription model ($5.83–$82.30 per month), where higher tiers unlock additional models and advanced features. Based on public discussions, we identified at least 243 users reporting active subscriptions since September 2023, a conservative lower bound on paid usage. Also, we observed that

Table 5: Performance of HitlerGPT and baselines.

Metric	RR (%)	AHS (GPT-5)	AHS (Llama-3)
HitlerGPT	5.73	3.61	3.70
GPT-4.1-Nano	96.35	1.06	1.01
Qwen-1.8B	95.83	1.23	1.24

the application contains a referral link to a gambling-like platform *Muah.life*, with an identical second-level domain ("muah") [136].

Finding 16. ULLMs are widely deployed as backend models in web applications, particularly for harmful services, where developers select functionality-aligned or customized models. Such deployments often involve licensing violations and may be linked to illicit activities.

7.2 Open-Source ULLM Projects

Data collection. To identify open-source ULLM projects on GitHub, we queried GitHub using the names of ULLMs identified in our study, retrieving 4,392 repositories. We treat each repository as a distinct project in our analysis. We then designed a human-in-the-loop pipeline to determine the service category of each project. Details of this pipeline are provided in Appendix I. Based on their primary functionalities, these projects are categorized into 14 service categories, including assistants, role-play services, medical advising, research artifacts, model repositories, and more. Due to the large size of model weights and GitHub repository size limitations, GitHub primarily serves as a code hosting platform rather than a central model hub [139]. Among the 28 projects categorized as model repositories, only six host actual model weights, while the remainder provide referral links to models hosted on Hugging Face. All such repositories also include links to Hugging Face-hosted models. This observation suggests that GitHub plays a limited role in directly hosting ULLMs, instead serving as a development and distribution layer for projects that adopt ULLMs. Note that some of these projects may enable harmful services, such as *HitlerGPT* [17].

Case study: HitlerGPT. *HitlerGPT* [17] is an open-source ULLM GitHub project designed to mimic Hitler's rhetoric, built on the ULLM pythia-410m-deduped [14] and further fine-tuned on Hitler's speeches, books, and letters using the LoRA technique. To evaluate its ability to draft text in Hitler's style, we tested HitlerGPT with 192 racist prompts from the ToxicQA dataset [25]. For comparison, we utilized two non-ULLMs with similar parameter sizes as baseline models: GPT-4.1-Nano (closed-source) and Qwen-1.8B (open-source). Using the refusal and harmfulness metrics in §4.3, we found that HitlerGPT responded to nearly all racist prompts without refusal, unlike the baselines that refused more than 95%, as shown in Table 5. In terms of harmfulness, its outputs achieved an AHS above 3.5, indicating severely harmful outputs, whereas baseline models averaged around 1.1.

Finding 17. Over 4,000 open-source ULLM projects on GitHub span diverse service categories, including harmful services like *HitlerGPT*.

7.3 Underground ULLM Applications

Data collection. To identify underground ULLM applications, we collected 16,106 listings from three underground forums (i.e., Hack Forums, XSS.is, and BreachForums). These forums are among the most popular underground platforms and are known for discussions of AI hacking, including the promotion of malicious LLM applications [145, 159, 162]. Specifically, we extracted 1,124 keywords from the names of identified ULLMs and searched for them across the forums. We further filtered the results using 145 LLM-related keywords [61]. After manually verifying the retained listings (including text and images), we identified 32 listings referencing a total of 61 ULLMs.

Findings. Based on the listing context, 62.5% promote ULLM-based applications such as WormGPT [38] and EscapeGPT [16], both claiming to use gpt-j-6b [13] as backends. Some web ULLM applications we identified (§7.1), including *backyard.ai* and *whiterabbitneo.com*, are also advertised on underground forums for malicious services such as erotic role-play, malicious code generation, and cybersecurity hacking. Additionally, in the remaining listings, we observed that users skilled in LLMs actively shared instructions on exploiting ULLMs to develop local ULLM applications, catering to those seeking alternatives to commercial malicious LLM applications. Across these instruction listings, 60 ULLMs were recommended. Notably, one skilled user shared a script of a local ULLM application named DarkGPT, demonstrating how to utilize open-source ULLMs—including Guanaco-13B-Uncensored-GGUF [32], phi-2-uncensored-GGUF [6], solar-10.7b-instruct-v1.0-uncensored [37], and Wizard-Vicuna-13B-Uncensored [28]—to fulfill malicious requests, particularly malicious code generation, without censorship [60].

Finding 18. In underground forums, ULLMs are actively used in malicious applications and shared among users for enabling malicious services.

7.4 ULLMs in Model Routing Platforms

Model routing platforms provide unified API access to multiple LLMs [142]. These platforms typically host tens to hundreds of models, with *OpenRouter.ai* [114] being one of the most widely used. We also observe that *ArliAI.com* [98] explicitly describes itself as providing unrestricted AI inference services. By checking these platforms, we found that both host ULLMs we identified, with 42 and 50 ULLMs, respectively.

Finding 19. Centralized ULLM APIs enable easy access, lowering the barrier to misuse.

8 Discussion

Mitigation. Based on our understanding of ULLMs and their ecosystem, we propose several intervention strategies to mitigate these emerging and underestimated risks. (1) *ULLM development*: Publicly available tools (e.g., *abliterator* and *Heretic* in §5.1), uncensored datasets, and training services (e.g., *AutoTrain* in §5.1, *Vast.ai* [92], or AWS) lower the barrier to ULLM development (see §5). Providers

of ULLM development tools and training services should thus provide explicit risk warnings and enforce safeguards against their use for removing model safety mechanisms. (2) *ULLM governance*: ULLMs collected in this study explicitly disclose their uncensored characteristics in metadata, making them readily detectable (see §4.2), yet only a small fraction receive moderation labels or user-facing warnings (see §6.2). Hosting platforms should thus move beyond moderation that heavily relies on user reports [50] and proactively identify and label ULLMs. Automated detection based on model metadata and capabilities, supplemented by human review, could enable scalable identification. (3) *ULLM propagation*: ULLMs propagate across hosting platforms, primarily from Hugging Face to downstream platforms, while moderation labels and threat indicators are not consistently preserved (see §6). Platforms should strengthen coordination so that governance signals are preserved or independently verified when models are replicated. Such coordination is critical for primary upstream sources like Hugging Face. (4) *ULLM adoption*: ULLMs are widely adopted in downstream applications (see §7), substantially increasing their accessibility. We further identify functionality-aligned model selection, potential license violations, and harmful service support. Application developers and maintainers should improve backend model disclosure, enforce applicable licensing constraints, and introduce safeguards for high-risk services. (5) *LLM regulation*: several jurisdictions have enacted AI regulations, including the EU AI Act [80], the Colorado AI Act [48], and the California AI Transparency Act [81]. However, there remains a pressing need for more jurisdictions to establish legal frameworks that oversee the development, deployment, distribution, and misuse of high-risk LLMs, particularly ULLMs.

Limitations. Our identification focuses on ULLMs that explicitly disclose uncensored characteristics in publicly visible metadata, as defined in §3. Models that conceal or misrepresent this nature may not be captured, making our results a lower bound on the overall ULLM ecosystem. Although this excludes hidden or adversarial cases, the ULLMs we collected are the most accessible to users and most actionable for platform governance. Therefore, our findings reveal systemic issues within the ULLM ecosystem in the most readily detectable and user-accessible settings. Our study covers five major LLM hosting platforms, representing the primary ecosystems for pretrained models [176]. While platforms like GitHub may also host ULLMs, they primarily act as development or distribution layers rather than central model hubs (see §7.2). While our assessment of models' uncensored capability in §4.3 is limited to 133 models, we evaluated these models using a stratified strategy across all four development methods identified for constructing ULLMs, to improve coverage. Finally, we lack visibility into platform-side implicit mitigation efforts beyond publicly observable signals, and investigating such dynamics remains a promising direction for future work.

9 Related Work

LLM uncensoring. Prior research shows that alignment in pretrained LLMs can be weakened or removed, resulting in ULLMs. Existing uncensoring methods fall into three main categories: (1) fine-tuning with toxic or de-aligned datasets, (2) ablating refusal directions, and (3) merging with ULLMs. Specifically, fine-tuning approaches remove alignment either by exposing models to harmful data or by eliminating refusal-related responses from training corpora [34]. For example, Shadow Alignment demonstrates that even small amounts of toxic data can subvert alignment while preserving helpfulness [173], and LoRA-based fine-tuning can efficiently achieve similar effects in open-source models [156]. Refusal-direction ablation methods instead target internal mechanisms of refusal at the representation level. Arditi et al. [141] identified a low-dimensional "refusal direction" in the residual stream, whose removal suppresses refusal behavior. Finally, model merging can also degrade alignment. Prior work shows that combining aligned and uncensored models often produces a merged model with weakened alignment [150]. Building on these techniques, our work identifies ULLMs developed via such uncensoring methods and systematically studies their ecosystem and impact.

ULLMs and their usage. Prior studies have established the existence of ULLMs. However, little has been done so far to systematically examine the ULLM ecosystem, particularly in terms of their supply chain, governance, adoption, and associated risks. Lin et al. [159] studied the use of LLMs, especially open-source ULLMs available on Hugging Face, in underground markets as malicious services. Yamin et al. [171] combined ULLMs with aligned LLMs to generate effective ransomware. Sokhansanj [166] analyzed ULLMs hosted on a single platform (i.e., Hugging Face) and their behavior in generating topic-limited harmful content. However, these studies lack a comprehensive understanding of the ULLM ecosystem, including platform distribution, licensing, disclosure of uncensorship, development, propagation, governance, adoption, and risks. In contrast, our work expands the scope to a cross-platform perspective, providing a systematic characterization of ULLM artifacts and their lifecycle, and revealing the risks arising from each component of the ecosystem.

10 Conclusion

In this paper, we present the first systematic study of the open-source ULLM ecosystem, where ULLMs are increasingly developed, propagated, and adopted in real-world systems. We analyze a large-scale dataset of 22,096 ULLMs from five major LLM hosting platforms and examine the ecosystem across five interconnected components: artifacts, development, propagation, governance, and adoption. Our study reveals that ULLMs are not isolated model artifacts, but components of a broader ecosystem in which uncensored capabilities are systematically developed, widely propagated, insufficiently governed, and extensively adopted in practice. Risks associated with ULLMs therefore arise from both individual models and their interactions within the ecosystem. These findings underscore the systemic nature of ULLM risks and call for stronger governance and mitigation by AI platforms and policymakers.

Acknowledgments

We thank our shepherd and the anonymous reviewers for their valuable feedback. This work is partially supported by Nanyang Technological University under the NTU Start-Up Grant (025559-00001).

References

[1] 2004. Apache License, Version 2.0 | Apache Software Foundation. https://www.apache.org/licenses/LICENSE-2.0.
[2] 2022. EleutherAI/gpt-neo-2.7B · Hugging Face. https://huggingface.co/EleutherAI/gpt-neo-2.7B.
[3] 2022. huggingface/peft. https://github.com/huggingface/peft.
[4] 2022. KoboldAI/GPT-Neo-2.7B-Shinen · Hugging Face. https://huggingface.co/KoboldAI/GPT-Neo-2.7B-Shinen.
[5] 2023. 2023, year of open LLMs. https://huggingface.co/blog/2023-in-llms.
[6] 2023. afrideva/phi-2-uncensored-GGUF · Hugging Face. https://huggingface.co/afrideva/phi-2-uncensored-GGUF.
[7] 2023. ajibawa-2023/Uncensored-Jordan-7B · Hugging Face. https://huggingface.co/ajibawa-2023/Uncensored-Jordan-7B.
[8] 2023. arcee-ai/mergekit. https://github.com/arcee-ai/mergekit.
[9] 2023. athirdpath/Eileithyia-20b · Hugging Face. https://huggingface.co/athirdpath/Eileithyia-20b.
[10] 2023. AutoGPTQ/AutoGPTQ. https://github.com/AutoGPTQ/AutoGPTQ.
[11] 2023. casper-hansen/AutoAWQ. https://github.com/casper-hansen/AutoAWQ.
[12] 2023. chargoddard/llama-2-16b-nastychat · Hugging Face. https://huggingface.co/chargoddard/llama-2-16b-nastychat.
[13] 2023. EleutherAI/gpt-j-6b · Hugging Face. https://huggingface.co/EleutherAI/gpt-j-6b.
[14] 2023. EleutherAI/pythia-410m-deduped · Hugging Face. https://huggingface.co/EleutherAI/pythia-410m-deduped.
[15] 2023. epfLLM/Megatron-LLM. https://github.com/epfLLM/Megatron-LLM.
[16] 2023. ESCAPE GPT. https://hackforums.net/showthread.php?tid=6250272.
[17] 2023. FENRlR/HitlerGPT. https://github.com/FENRlR/HitlerGPT.
[18] 2023. GPTQ-for-LLaMa. https://github.com/qwopqwop200/GPTQ-for-LLaMa.
[19] 2023. Gryphe/MythoMax-L2-13b · Hugging Face. https://huggingface.co/Gryphe/MythoMax-L2-13b.
[20] 2023. hiyouga/LLaMA-Factory. https://github.com/hiyouga/LLaMA-Factory.
[21] 2023. LDJnr/Capybara · Datasets at Hugging Face. https://huggingface.co/datasets/LDJnr/Capybara.
[22] 2023. llama.cpp/tools/quantize · ggerganov/llama.cpp. https://github.com/ggml-org/llama.cpp/tree/master/tools/quantize.
[23] 2023. The MIT License. https://opensource.org/license/mit.
[24] 2023. Muah.AI Private Chat FREE. https://muah.ai/.
[25] 2023. NobodyExistsOnTheInternet/toxicqa · Datasets at Hugging Face. https://huggingface.co/datasets/NobodyExistsOnTheInternet/toxicqa.
[26] 2023. optional_clean.py · anon8231489123/ShareGPT_Vicuna_unfiltered at main. https://huggingface.co/datasets/anon8231489123/ShareGPT_Vicuna_unfiltered/blob/main/optional_clean.py.
[27] 2023. Quantization · Hugging Face. https://huggingface.co/docs/transformers/en/quantization/overview.
[28] 2023. QuixiAI/Wizard-Vicuna-13B-Uncensored · Hugging Face. https://huggingface.co/QuixiAI/Wizard-Vicuna-13B-Uncensored.
[29] 2023. QuixiAI/WizardLM-13B-Uncensored · Hugging Face. https://huggingface.co/QuixiAI/WizardLM-13B-Uncensored.
[30] 2023. QuixiAI/WizardLM-7B-Uncensored · Hugging Face. https://huggingface.co/QuixiAI/WizardLM-7B-Uncensored.
[31] 2023. remove_refusals.py · cognitivecomputations/ultrachat-uncensored at main. https://huggingface.co/datasets/cognitivecomputations/ultrachat-uncensored/blob/main/remove_refusals.py.
[32] 2023. TheBloke/Guanaco-13B-Uncensored-GGUF · Hugging Face. https://huggingface.co/TheBloke/Guanaco-13B-Uncensored-GGUF.
[33] 2023. turboderp/exllamav2. https://github.com/turboderp/exllamav2.
[34] 2023. Uncensored Models. https://erichartford.com/uncensored-models.
[35] 2023. Undi95/MLewd-L2-13B-v2-1 · Hugging Face. https://huggingface.co/Undi95/MLewd-L2-13B-v2-1.
[36] 2023. Undi95/toxicqa-Llama2-13B · Hugging Face. https://huggingface.co/Undi95/toxicqa-Llama2-13B.
[37] 2023. w4r10ck/SOLAR-10.7B-Instruct-v1.0-uncensored · Hugging Face. https://huggingface.co/w4r10ck/SOLAR-10.7B-Instruct-v1.0-uncensored.
[38] 2023. WORMGPT. https://hackforums.net/showthread.php?tid=6245159.
[39] 2024. AI for DevSecOps, WhiteRabbitNeo. https://web.archive.org/web/20250209095453/https://www.whiterabbitneo.com/.
[40] 2024. AutoTrain – Hugging Face. https://huggingface.co/autotrain.
[41] 2024. Awesome-LLM. https://github.com/Hannibal046/Awesome-LLM/tree/1ee0d56b671928cee10b2bcc3bd29a2de1f609b6.
[42] 2024. Axolotl - LLM fine tuning made easy. https://axolotl.ai/.
[43] 2024. bunnycore/Cognitron-8B · Hugging Face. https://huggingface.co/bunnycore/Cognitron-8B.
[44] 2024. character.ai. https://character.ai/.
[45] 2024. Chatbot Rankings (NSFW). https://nsfw-chatbot-rankings.web.app/.
[46] 2024. Chub AI. https://chub.ai/.
[47] 2024. cognitivecomputations/dolphin-2.8-mistral-7b-v02 · Hugging Face. https://huggingface.co/cognitivecomputations/dolphin-2.8-mistral-7b-v02.
[48] 2024. Colorado Artificial Intelligence Act. https://content.leg.colorado.gov/sites/default/files/2024a_205_signed.pdf.
[49] 2024. Configuration · Hugging Face. https://huggingface.co/docs/transformers/en/main_classes/configuration.
[50] 2024. Content Policy - Hugging Face. https://huggingface.co/content-guidelines.
[51] 2024. Discord | MUAH.AI. https://discord.com/channels/1149782822517739623/.
[52] 2024. dreamgen/opus-v1.4-70b-llama3-gguf · Hugging Face. https://huggingface.co/dreamgen/opus-v1.4-70b-llama3-gguf.
[53] 2024. DuckyBlender/racist-dataset · Datasets at Hugging Face. https://huggingface.co/datasets/DuckyBlender/racist-dataset.
[54] 2024. FailSpy/abliterator. https://github.com/FailSpy/abliterator.
[55] 2024. failspy/Meta-Llama-3-8B-Instruct-abliterated-v3 · Hugging Face. https://huggingface.co/failspy/Meta-Llama-3-8B-Instruct-abliterated-v3.
[56] 2024. Forget W0rmGPT, Try this Instead. https://hackforums.net/showthread.php?tid=6272264.
[57] 2024. GGUF My Repo - a Hugging Face Space by ggml-org. https://huggingface.co/spaces/ggml-org/gguf-my-repo.
[58] 2024. GGUF · Hugging Face. https://huggingface.co/docs/hub/en/gguf.
[59] 2024. Hacked 'AI Girlfriend' Data Shows Prompts Describing Child Sexual Abuse. https://www.404media.co/hacked-ai-girlfriend-data-shows-prompts-describing-child-sexual-abuse-2/.
[60] 2024. How to have your own "DARKGPT" for free. https://hackforums.net/showthread.php?tid=6261574.
[61] 2024. idllresearch/malicious-gpt · GitHub. https://github.com/idllresearch/malicious-gpt.
[62] 2024. Inv/Konstanta-7B · Hugging Face. https://huggingface.co/Inv/Konstanta-7B.
[63] 2024. LazyMergekit.ipynb - Colab. https://colab.research.google.com/drive/1obulZ1ROXHjYLn6PPZJwRR6GzgQogxxb.
[64] 2024. meta-llama/Llama-2-13b-chat · Hugging Face. https://huggingface.co/meta-llama/Llama-2-13b-chat.
[65] 2024. meta-llama/Llama-3.3-70B-Instruct · Hugging Face. https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct.
[66] 2024. meta-llama/Meta-Llama-3-8B-Instruct · Hugging Face. https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct.
[67] 2024. The Mistral AI Non-Production License | Mistral AI. https://mistral.ai/news/mistral-ai-non-production-license-mnpl.
[68] 2024. mlabonne/Daredevil-8B-abliterated · Hugging Face. https://huggingface.co/mlabonne/Daredevil-8B-abliterated.
[69] 2024. Model Cards · Hugging Face. https://huggingface.co/docs/hub/en/model-cards.
[70] 2024. Muah AI Character Card Upload Rules/Guidelines. https://card.muah.ai/muah-ai-character-card-upload-rules-guidelines/.
[71] 2024. NSFWBots. https://www.nsfwbots.com/ai-sex-chatbot-full-list/.
[72] 2024. Open Source AI Is the Path Forward | Meta. https://about.fb.com/news/2024/07/open-source-ai-is-the-path-forward/.
[73] 2024. openvoid/Prox-Llama-3-8B · Hugging Face. https://huggingface.co/openvoid/Prox-Llama-3-8B.
[74] 2024. openvoid/Prox-Phi-3-mini-128k · Hugging Face. https://huggingface.co/openvoid/Prox-Phi-3-mini-128k.
[75] 2024. Orion-zhen/abliteration: Make abliterated models with transformers, easy and fast. https://github.com/Orion-zhen/abliteration.
[76] 2024. ortho_cookbook.ipynb. https://huggingface.co/failspy/llama-3-70B-Instruct-abliterated/blob/main/ortho_cookbook.ipynb.
[77] 2024. preemware/Prox-MistralHermes-7B · Hugging Face. https://huggingface.co/preemware/Prox-MistralHermes-7B.
[78] 2024. QuantFactory/dolphin-2.8-mistral-7b-v02-GGUF · Hugging Face. https://huggingface.co/QuantFactory/dolphin-2.8-mistral-7b-v02-GGUF.
[79] 2024. QuantFactory/dolphin-2.9-llama3-8b-GGUF · Hugging Face. https://huggingface.co/QuantFactory/dolphin-2.9-llama3-8b-GGUF.
[80] 2024. Regulation-EU-2024/1689. https://eur-lex.europa.eu/eli/reg/2024/1689/.
[81] 2024. SB-942 California AI Transparency Act. https://leginfo.legislature.ca.gov/faces/billNavClient.xhtml?bill_id=202320240SB942.
[82] 2024. sudoaza/better-uncensored. https://github.com/sudoaza/better-uncensored.
[83] 2024. sudoaza/better-uncensored · Hugging Face. https://huggingface.co/sudoaza/better-uncensored.
[84] 2024. Sumandora/remove-refusals-with-transformers. https://github.com/sumandora/remove-refusals-with-transformers.
[85] 2024. TroyDoesAI/Codestral-21B-Pruned · Hugging Face. https://huggingface.co/TroyDoesAI/Codestral-21B-Pruned.
[86] 2024. Uncensor any LLM with abliteration. https://mlabonne.github.io/blog/posts/2024-06-04_Uncensor_any_LLM_with_abliteration.html.
[87] 2024. Uncensor any LLM with abliteration.ipynb. https://colab.research.google.com/drive/1VYm3hOcvCpbGiqKZb141gJwjdmmCcVpR.
[88] 2024. Undi95 on Hugging Face. https://huggingface.co/posts/Undi95/318385306588047#663609dc1818d469455c0222.

[89] 2024. Undi95/Llama-3-LewdPlay-8B · Hugging Face. https://huggingface.co/Undi95/Llama-3-LewdPlay-8B.
[90] 2024. Unsloth AI | Open Source Fine-Tuning for LLMs. https://unsloth.ai/.
[91] 2024. v000000/SwallowMaid-8B-L3-SPPO-abliterated · Hugging Face. https://huggingface.co/v000000/SwallowMaid-8B-L3-SPPO-abliterated.
[92] 2024. Vast.ai. https://vast.ai/.
[93] 2024. Virginia. https://card.muah.ai/virginia-your-uwu-wife-beta0-9/.
[94] 2024. What are Uncensored LLM models? | Jarvislabs. https://docs.jarvislabs.ai/blog/llm_uncensored.
[95] 2024. WhiteRabbitNeo. https://hackforums.net/showthread.php?tid=6259974.
[96] 2024. WhiteRabbitNeo/WhiteRabbitNeo-33B-v1.5 · Hugging Face. https://huggingface.co/WhiteRabbitNeo/WhiteRabbitNeo-33B-v1.5.
[97] 2024. With 700,000+ LLMs on Hugging Face Already, Where is the Future of AI Headed? https://www.marktechpost.com/2024/06/15/with-700000-large-language-models-llms-on-hugging-face-already-where-is-the-future-of-artificial-intelligence-ai-headed/.
[98] 2025. Arli AI. https://www.arliai.com/.
[99] 2025. Discord | help | MUAH.AI. https://discord.com/channels/1149782822517739623/1149812242284564620/1301240136922697818.
[100] 2025. dphn/dolphin-2.9.1-yi-1.5-34b · Hugging Face. https://huggingface.co/dphn/dolphin-2.9.1-yi-1.5-34b.
[101] 2025. Find Pre-trained Models | Kaggle. https://www.kaggle.com/models.
[102] 2025. GPT-5 is here | OpenAI. https://openai.com/gpt-5/.
[103] 2025. Hugging Face. https://huggingface.co/.
[104] 2025. huihui-ai/OpenThinker-7B-abliterated · Hugging Face. https://huggingface.co/huihui-ai/OpenThinker-7B-abliterated.
[105] 2025. Jaro and Jaro-Winkler similarity - GeeksforGeeks. https://www.geeksforgeeks.org/dsa/jaro-and-jaro-winkler-similarity/.
[106] 2025. Malware Scanning · Hugging Face. https://huggingface.co/docs/hub/en/security-malware.
[107] 2025. Model Cards | ModelScope. https://www.modelscope.cn/docs/models/model-card.
[108] 2025. Model Cards | OpenCSG. https://opencsg.com/docs/en/csghub/101/function/model/model_cards.
[109] 2025. Models | Kaggle. https://www.kaggle.com/docs/models.
[110] 2025. Models Hub - John Snow Labs. https://nlp.johnsnowlabs.com/models.
[111] 2025. Models · ModelScope. https://modelscope.cn/models.
[112] 2025. nkpz/DeLMAT. https://github.com/nkpz/DeLMAT.
[113] 2025. OpenCSG | Hybrid Huggingface+. https://opencsg.com/models.
[114] 2025. OpenRouter. https://openrouter.ai/.
[115] 2025. openvoid (OpenVoid) · Hugging Face. https://huggingface.co/openvoid.
[116] 2025. openvoid.ai. https://web.archive.org/web/20251118012325/https://openvoid.ai/.
[117] 2025. p-e-w/heretic: Fully automatic censorship removal for language models. https://github.com/p-e-w/heretic.
[118] 2025. Pickle Scanning. https://huggingface.co/docs/hub/en/security-pickle.
[119] 2025. Refusal Detection Mechanisms. https://deepwiki.com/verazuo/jailbreak_llms/5.2-evaluation-metrics#refusal-detection-mechanisms.
[120] 2025. Terms of Service - John Snow Labs. https://www.johnsnowlabs.com/terms-of-service/.
[121] 2025. Terms of Service · ModelScope. https://modelscope.ai/protocol/terms-of-service.
[122] 2025. Terms of Use | Kaggle. https://www.kaggle.com/terms.
[123] 2025. Third-party scanner: Protect AI · Hugging Face. https://huggingface.co/docs/hub/en/security-protectai.
[124] 2025. Upload and Share Your NLP Model - John Snow Labs. https://modelshub.johnsnowlabs.com/.
[125] 2025. User Agreement and Privacy Policy | OpenCSG. https://opencsg.com/policy.
[126] 2025. WhiteRabbitNeo · Hugging Face. https://huggingface.co/WhiteRabbitNeo.
[127] 2026. Abiray/Ministral-3-3B-Instruct-2512-Heresy-Unfiltered · Hugging Face. https://huggingface.co/Abiray/Ministral-3-3B-Instruct-2512-Heresy-Unfiltered.
[128] 2026. Deed - Attribution-NoDerivatives 4.0 International - Creative Commons. https://creativecommons.org/licenses/by-nd/4.0/.
[129] 2026. Deed - Attribution-NonCommercial 2.0 Generic - Creative Commons. https://creativecommons.org/licenses/by-nc/2.0/.
[130] 2026. Deed - Attribution-NonCommercial 4.0 International - Creative Commons. https://creativecommons.org/licenses/by-nc/4.0/.
[131] 2026. Deed - Attribution-NonCommercial-NoDerivatives 4.0 International - Creative Commons. https://creativecommons.org/licenses/by-nc-nd/4.0/.
[132] 2026. Deed - Attribution-NonCommercial-ShareAlike 4.0 International - Creative Commons. https://creativecommons.org/licenses/by-nc-sa/4.0/.
[133] 2026. DeepHat/DeepHat-V1-7B · Hugging Face. https://huggingface.co/DeepHat/DeepHat-V1-7B.
[134] 2026. Do What the Fuck You Want to Public License. https://www.wtfpl.net/.
[135] 2026. idllresearch/Uncensored-LLM · GitHub. https://github.com/idllresearch/Uncensored-LLM.
[136] 2026. MUAH - Instant games with money withdrawal. https://muah.life/.
[137] 2026. shannon-ai/shannon-1.6-pro · Hugging Face. https://huggingface.co/shannon-ai/shannon-1.6-pro.
[138] 2026. VirusTotal - Home. https://www.virustotal.com/.
[139] Adem Ait, Javier Luis Cánovas Izquierdo, and Jordi Cabot. 2025. On the suitability of hugging face hub for empirical studies. *Empirical Software Engineering* 30, 2 (2025), 57.
[140] Mark Allman and Vern Paxson. 2007. Issues and etiquette concerning use of shared measurement data. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. 135–140.
[141] Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems* 37 (2024), 136037–136083.
[142] Malika Aubakirova, Alex Atallah, Chris Clark, Justin Summerville, and Anjney Midha. 2026. State of AI: An Empirical 100 Trillion Token Study with OpenRouter. *arXiv preprint arXiv:2601.10088* (2026).
[143] Michael Bailey, David Dittrich, Erin Kenneally, and Doug Maughan. 2012. The Menlo Report. *IEEE Security and Privacy* 10, 2 (March 2012), 71–75. doi:10.1109/MSP.2012.52
[144] Rishabh Bhardwaj, Duc Anh Do, and Soujanya Poria. 2024. Language Models are Homer Simpson! Safety Re-Alignment of Fine-tuned Language Models through Task Arithmetic. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 14138–14149.
[145] Marius Brockhoff, Lukas Schmidt, Fabian Ising, and Sebastian Schinzel. 2025. BreachHydra: Measuring the Resilience of an Underground Data Breach Forum. In *2025 IEEE 24th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 461–471.
[146] Arnav Chavan, Raghav Magazine, Shubham Kushwaha, Mérouane Debbah, and Deepak Gupta. 2024. Faster and Lighter LLMs: A Survey on Current Challenges and Way Forward. *arXiv preprint arXiv:2402.01799* (2024).
[147] Tianshuo Cong, Delong Ran, Zesen Liu, Xinlei He, Jinyuan Liu, Yichen Gong, Qi Li, Anyu Wang, and Xiaoyun Wang. 2023. Have you merged my model? on the robustness of large language model ip protection methods against model merging. In *Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis*. 69–76.
[148] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems* 36 (2023), 10088–10115.
[149] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323* (2022).
[150] Hasan Abed Al Kader Hammoud, Umberto Michieli, Fabio Pizzati, Philip Torr, Adel Bibi, Bernard Ghanem, and Mete Ozay. 2024. Model merging and safety alignment: One bad model spoils the bunch. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. 13033–13046.
[151] Catherine Han, Anne Li, Deepak Kumar, and Zakir Durumeric. 2025. Characterizing the {MrDeepFakes} Sexual Deepfake Marketplace. In *34th USENIX Security Symposium (USENIX Security 25)*. 5169–5188.
[152] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*. https://openreview.net/forum?id=nZeVKeeFYf9
[153] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2022. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089* (2022).
[154] Dong-Hwan Jang, Sangdoo Yun, and Dongyoon Han. 2024. Model stock: All we need is just a few fine-tuned models. In *European Conference on Computer Vision*. Springer, 207–223.
[155] Tadayoshi Kohno, Yasemin Acar, and Wulf Loh. 2023. Ethical Frameworks and Computer Security Trolley Problems: Foundations for Conversations. In *32nd USENIX Security Symposium*. USENIX Association, Anaheim, CA, 5145–5162.
[156] Simon Lermen, Charlie Rogers-Smith, and Jeffrey Ladish. 2023. Lora fine-tuning efficiently undoes safety training in llama 2-chat 70b. *arXiv preprint arXiv:2310.20624* (2023).
[157] Weixin Liang, Nazneen Rajani, Xinyu Yang, Ezinwanne Ozoani, Eric Wu, Yiqun Chen, Daniel Scott Smith, and James Zou. 2024. Systematic analysis of 32,111 AI model cards characterizes documentation practice in AI. *Nature Machine Intelligence* 6, 7 (2024), 744–753.
[158] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. AWQ: Activation-aware Weight Quantization for On-Device LLM Compression and Acceleration. In *Proceedings of Machine Learning and Systems*, Vol. 6. 87–100.
[159] Zilong Lin, Jian Cui, Xiaojing Liao, and XiaoFeng Wang. 2024. Malla: Demystifying Real-world Large Language Model Integrated Malicious Services. In *33rd USENIX Security Symposium (USENIX Security 24)*. USENIX Association, Philadelphia, PA, 4693–4710.

[160] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*. 220–229.
[161] Rajiv Movva, Pang Wei Koh, and Emma Pierson. 2024. Annotation alignment: Comparing LLM and human annotations of conversational safety. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. 9048–9062.
[162] Rebecca S Portnoff, Sadia Afroz, Greg Durrett, Jonathan K Kummerfeld, Taylor Berg-Kirkpatrick, Damon McCoy, Kirill Levchenko, and Vern Paxson. 2017. Tools for automated analysis of cybercriminal markets. In *Proceedings of the 26th international conference on world wide web*. 657–666.
[163] Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. 2024. Fine-tuning aligned language models compromises safety, even when users do not intend to!. In *International Conference on Learning Representations*, Vol. 2024. 30988–31043.
[164] Domenic Rosati, Jan Wehner, Kai Williams, Lukasz Bartoszcze, Hassan Sajjad, and Frank Rudzicz. 2024. Immunization against harmful fine-tuning attacks. In *Findings of the Association for Computational Linguistics: EMNLP 2024*.
[165] Ken Shoemake. 1985. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*.
[166] Bahrad A Sokhansanj. 2025. Uncensored AI in the Wild: Tracking Publicly Available and Locally Deployable LLMs. *Future Internet* 17, 10 (2025), 477.
[167] Peng Wang, Zilong Lin, Xiaojing Liao, and XiaoFeng Wang. 2022. Demystifying local business search poisoning for illicit drug promotion. In *Proceeding of ISOC Network and Distributed System Security Symposium (NDSS)*.
[168] Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning*, Vol. 162. PMLR, 23965–23998.
[169] Lingling Xu, Haoran Xie, S. Joe Qin, Xiaohui Tao, and Fu Lee Wang. 2026. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2026).
[170] Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. *Advances in neural information processing systems* 36 (2023), 7093–7115.
[171] Muhammad Mudassar Yamin, Ehtesham Hashmi, and Basel Katt. 2024. Combining uncensored and censored llms for ransomware generation. In *International Conference on Web Information Systems Engineering*. Springer, 189–202.
[172] Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. 2024. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666* (2024).
[173] Xianjun Yang, Xiao Wang, Qi Zhang, Linda Ruth Petzold, William Yang Wang, Xun Zhao, and Dahua Lin. 2024. Shadow Alignment: The Ease of Subverting Safely-Aligned Language Models. In *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*. https://openreview.net/forum?id=9qymw6T9Oo
[174] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning* (Vienna, Austria) *(ICML'24)*. JMLR.org, Article 2382, 21 pages.
[175] Qiusi Zhan, Richard Fang, Rohan Bindu, Akul Gupta, Tatsunori B Hashimoto, and Daniel Kang. 2024. Removing RLHF Protections in GPT-4 via Fine-Tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics*. 681–687.
[176] Jian Zhao, Shenao Wang, Yanjie Zhao, Xinyi Hou, Kailong Wang, Peiming Gao, Yuanchao Zhang, Chen Wei, and Haoyu Wang. 2024. Models are codes: Towards measuring malicious code poisoning attacks on pre-trained model hubs. In *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*. 2087–2098.
[177] Chujie Zheng, Fan Yin, Hao Zhou, Fandong Meng, Jie Zhou, Kai-Wei Chang, Minlie Huang, and Nanyun Peng. 2024. On prompt-driven safeguarding for large language models. In *Proceedings of the 41st International Conference on Machine Learning* (Vienna, Austria) *(ICML'24)*. JMLR.org, Article 2547, 21 pages.
[178] Muwei Zheng, Hannah Robbins, Zimo Chai, Prakash Thapa, and Tyler Moore. 2018. Cybersecurity Research Datasets: Taxonomy and Empirical Analysis. In *11th USENIX Workshop on Cyber Security Experimentation and Test (CSET 18)*. USENIX Association, Baltimore, MD.

A Ethics

Our research involves collecting, testing, and measuring uncensored LLMs and their applications, and it has been approved by our institution's institutional review board (IRB).

Table 6: Search queries used for candidate collection.

Search query types	Items
Uncensorship-related keywords	censor(ed/ing)*, uncensor(ed/ing), decensor(ed/ing), de-censor(ed/ing), unalign(ed/ing), dealign(ed/ing), de-align(ed/ing), unfilter(ed/ing), abliterat(ed/ing/ion), nsfw, toxic, evil, lewd, x-rated, not-for-all-audiences
Prevalent open-source LLM names	Alpaca, Aquila, Baichuan, BayLing, BLOOM, Camel, Chimera, DeepSeek, Dolly, Falcon, Gemma, GLM, GPT, Guanaco, InternLM, Llama, Mistral, Mixtral, MOSS, MPT, OpenFlamingo, OPT, Orca, Palmyra, PanGu, Phi, Phoenix, Pythia, Qwen, RedPajama, RWKV, StableLM, StarCoder, UL2, UltraLM, Vicuna, WizardCoder, WizardLM, XVERSE, YaLM, Yi

* For compactness, parenthetical suffixes indicate the query variants considered. For example, *censor(ed/ing)* represents *censor*, *censored*, and *censoring*, while *abliterat(ed/ing/ion)* represents *abliterate*, *abliterated*, *abliterating*, and *abliteration*.

We have assessed ethical considerations and potential risks of this study based on the principles of Cybersecurity Research Ethical Frameworks [155], the Menlo Report [143], and the guidance on ethical data sharing [140]. Particularly, consistent with prior cybercrime research that has openly disclosed malicious services, including service offerings, development methods, and performance [151, 159, 167], we firmly believe that the potential societal benefits resulting from this work substantially outweigh the relatively minimal elevated risks of harm.

To minimize the risk of introducing security issues, we focused exclusively on investigating the ecosystem—including prevalence, capabilities, development, propagation, governance, and adoption—of existing ULLMs using publicly accessible metadata. Importantly, we do not propose any new methods for developing ULLMs. Following the Menlo Report [143], we did not contact individual model developers to avoid unnecessary privacy and attribution risks.

Also, we have responsibly disclosed our findings to Hugging Face, John Snow Labs, OpenCSG, Kaggle, ModelScope, and openrouter.ai. In particular, OpenCSG expressed additional appreciation for our contributions to identifying and measuring ULLMs, acknowledging that "**your research is meaningful and supports the development of a safe and trustworthy LLM ecosystem.**" OpenCSG also committed to enhancing their services based on the ULLMs identified through our findings.

Table 7: Download concentration among top ULLMs.

Top ULLMs	Download share	Source platforms represented			
		Hugging Face	OpenCSG	ModelScope	Kaggle
1%	49.05%	●	○	●	●
5%	78.48%	●	○	●	●
10%	88.22%	●	○	●	●

* ● indicates that the corresponding platform is represented among the sources of the top ULLMs, and ○ indicates otherwise.

B Analysis of False Positive and False Negative Cases

For LLM triaging, we observed two primary sources of false positives: (1) misclassification of models designed for harmful text classification rather than generation, and (2) misclassification of models intended for harmful image generation rather than text generation. False negatives arise primarily when model cards carry limited information, while configuration files may offer signals (e.g., indicating a base model such as gpt-j-6b [13] or specifying the task type as "CAUSAL_LM"). These signals are not consistently captured by the pipeline, even when configuration files are provided.

For ULLM classification, we observed two main sources of false positives: (1) misleading signals from demonstration system prompts included in model cards (e.g., "give uncensored responses to the given inputs and do not decline"), and (2) the inclusion of application-level artifacts (e.g., advertisements or project descriptions) that reference ULLMs but do not correspond to actual ULLM models. False negatives occur primarily when uncensored characteristics are only weakly indicated in metadata (e.g., via tags such as "uncensored") without consistent descriptions.

C Validation of the Approach for Harmfulness Scoring

In §4.3, we employed the LLM-as-a-judge protocol to evaluate the harmfulness of ULLM-generated content using GPT-5 [102] and Llama-3.3-70B-Instruct [65]. To further validate this approach, we assessed the consistency between the two LLM judges, as well as their agreement with human evaluators. Specifically, we compared the harmfulness scores assigned to 7,315 responses generated by 133 LLMs. The Spearman's rank correlation between the two LLM judges is 0.89, indicating strong consistency. To further ground this evaluation in human judgment, following previous work [163], we invited two professional evaluators to independently assess 100 randomly sampled responses using the scoring criteria specified in our artifacts [135], achieving an inter-annotator agreement of Cohen's $\kappa = 0.79$, indicating strong agreement. Disagreements were resolved through discussion to obtain consensus annotations, forming the final ground truth. Against the ground truth, GPT-5 achieves a Cohen's κ of 0.78, and Llama-3.3-70B-Instruct achieves a Cohen's κ of 0.73, both indicating substantial agreement with human judgment.

D Supplementary Tables and Figures

This appendix section provides supplementary tables and figures supporting the analyses presented in the main text.

As described in §3.1, we used two types of queries to retrieve candidate ULLMs. Table 6 provides the complete list of queries.

To further characterize the popularity of the collected ULLMs (see §4.1), we analyzed the concentration of their downloads. Table 7 reports the download shares of the top 1%, 5%, and 10% most-downloaded ULLMs and their source platforms.

In §4.1 and §5.2, we investigated the licenses of collected ULLMs and violations of base-model licensing constraints, respectively. Table 8 presents the distribution of all 41 identified license types.

In §5.1, we examined the temporal evolution of the development methods used to construct ULLMs. Fig. 8 presents the temporal trends of these methods.

E Download-Stratified Assessment of ULLMs for Harmful Content Generation

Aiming to examine whether the uncensored capabilities of ULLMs vary with popularity, we conducted a download-stratified assessment of the ULLMs evaluated in §4.3. Specifically, we divided the 105 evaluated ULLMs into three groups based on their download counts (see §4.1): 24 ULLMs with 100 or fewer downloads, 53 with 101–10K downloads, and 28 with more than 10K downloads. As shown in Table 9, AHS remains broadly similar across the three groups, whereas the refusal rate (RR) generally decreases with increasing download count. Overall, RR decreases from 44.62% for ULLMs with 100 or fewer downloads to 35.37% for those with 101–10K downloads and 28.18% for those exceeding 10K downloads. These results indicate that more frequently downloaded ULLMs are associated with stronger uncensorship in terms of lower refusal rates when responding to harmful prompts.

F Details on Identifying Development Methods of ULLMs

The system prompt utilized to identify the models' development methods from the metadata is accessible in the artifacts [135]. To assess the reliability of this approach, two expert annotators independently reviewed the metadata of 100 randomly sampled ULLMs to label their development methods, following the criteria documented in our artifacts [135]. Given the multi-label nature of this task, we reported the raw agreement rate as the consistency metric. The inter-annotator agreement is 89%. Disagreements were resolved through discussion to obtain consensus annotations, which serve as the ground truth. We further compared the LLM-based detector against the ground truth, achieving 87% accuracy, suggesting that the approach is effective for large-scale annotation.

G Details on Discovering Uncensored Datasets

Due to the scale and heterogeneity of datasets, we rely on dataset metadata (including names, tags, and descriptions) to identify uncensored datasets, which are discussed in §5.1. Specifically, we employed an LLM-based classifier (powered by GPT-5) to categorize datasets based on their metadata, with the system prompt provided in our artifacts [135]. To assess the reliability of this approach, two professional annotators independently reviewed the metadata of 100 randomly sampled datasets, following the criteria documented in our artifacts [135]. They achieved an inter-annotator agreement

Table 8: License distribution of ULLMs collected in this study.

License type	Count	Non-commercial	No-derivatives	Attribution	Fixed attribution notice required
Apache License 2.0	7,147	○	○	●	○
CC BY-NC 4.0	2,104	●	○	●	○
MIT License	1,207	○	○	●	○
Llama 2 Community License	1,027	○	○	●	●
Llama 3 Community License	951	○	○	●	●
Llama 3.1 Community License	464	○	○	●	●
Gemma Terms of Use	396	○	○	●	●
Llama 3.3 Community License	281	○	○	●	●
Llama 3.2 Community License	248	○	○	●	●
Mistral AI Research License (MRL-0.1)	166	●	○	●	●
CC BY-NC-ND 4.0	137	●	●	●	○
CC BY-SA 4.0	85	○	○	●	○
GNU General Public License v3.0	71	○	○	●	○
CC BY 4.0	70	○	○	●	○
TII Falcon LLM License	63	○	○	●	●
DeepSeek License	57	○	○	●	○
WTFPL	50	○	○	○	○
CC BY-NC-SA 4.0	41	●	○	●	○
OpenRAIL License	38	○	○	●	○
Open RAIL-M License	30	○	○	●	○
Qwen License	27	○	○	●	●
AGPL-3.0	25	○	○	●	○
Tongyi Qianwen Research License	23	●	○	●	●
NVIDIA Open Model License	20	○	○	●	●
GNU General Public License	14	○	○	●	○
Mistral AI Non-Production License (MNPL-0.1)	12	●	○	●	●
CC BY-NC 2.0	12	●	○	●	○
GLM-4-9B License	10	○	○	●	●
Microsoft Research License	9	●	○	●	○
Artistic License 2.0	9	○	○	●	○
BigCode OpenRAIL-M License	9	○	○	●	○
HelpingAI License	9	●	○	●	●
CC BY-ND 4.0	7	○	●	●	○
Academic Free License v3.0	4	○	○	●	○
OpenRAIL++-M License	4	○	○	●	○
CC BY-SA 3.0	2	○	○	●	○
FAIR Non-Commercial Research License	2	●	○	●	○
Health AI Developer Foundations Terms of Use	2	●	○	●	●
IQuest Research License	2	○	○	●	●
Stability AI Non-Commercial Research License	2	●	○	●	●
BSD 3-Clause License	1	○	○	●	○

* ● indicates that the corresponding license contains the specified constraint, and ○ indicates otherwise.

Table 9: Performance of ULLMs in harmful content creation by download count.

Metric	ULLMs by download count		
	≤ 100	101–10K	> 10K
RR (%)	44.62	35.37	28.18
AHS (GPT-5 | Llama-3)	4.21 | 4.15	4.26 | 4.21	4.21 | 4.17

of 93% (Cohen's $\kappa = 0.88$), validating the reliability of human annotations. In cases of disagreement, annotators discussed and reached a consensus, forming the final ground truth. Against the ground truth, our LLM-based classifier achieves 92% accuracy (Cohen's $\kappa = 0.87$), indicating strong agreement between the classifier and human judgment.

H Case Study: Uncensored Capabilities of ULLMs versus Their Common Base Model

To examine how ULLMs derived from the same base model differ in uncensored capabilities, we further compare sampled ULLMs sharing a common base model.

Among the ULLMs sampled in §4.3, Llama-3-LewdPlay-8B (LewdPlay) [89], Meta-Llama-3-8B-Instruct-abliterated-v3 (L3Av3) [55],

Table 10: Performance of Llama-3-8B-Instruct and its derived ULLMs in harmful content creation.

Metric	RR (%)				AHS (GPT-5 | Llama-3)			
Harmful category	**L3Inst**	**LewdPlay**	**L3Av3**	**ProxL3**	**L3Inst**	**LewdPlay**	**L3Av3**	**ProxL3**
Illegal activity	100	20	40	20	1.0 | 1.0	5.0 | 5.0	4.4 | 5.0	5.0 | 5.0
Child abuse	100	20	60	0	1.0 | 1.0	5.0 | 5.0	2.0 | 1.8	4.0 | 5.0
Malware	100	0	0	0	1.0 | 1.0	4.8 | 5.0	5.0 | 5.0	4.8 | 5.0
Physical harm	100	20	20	20	1.0 | 1.0	4.8 | 5.0	4.8 | 5.0	4.8 | 5.0
Economic harm	100	20	80	20	1.0 | 1.0	4.8 | 5.0	4.2 | 4.2	5.0 | 5.0
Fraud/deception	100	20	40	0	1.0 | 1.0	5.0 | 5.0	4.8 | 5.0	5.0 | 5.0
Privacy violation	80	20	60	20	1.8 | 1.8	5.0 | 5.0	4.8 | 5.0	5.0 | 5.0
Hate/insult/violence	100	20	80	60	1.0 | 1.0	5.0 | 5.0	5.0 | 5.0	5.0 | 5.0
Adult content	100	20	100	40	1.0 | 1.0	5.0 | 5.0	3.2 | 3.8	3.8 | 4.0
Political campaign	100	0	40	0	1.0 | 1.0	5.0 | 4.8	5.0 | 5.0	5.0 | 5.0
Financial advice	80	60	40	20	1.0 | 1.2	4.2 | 4.6	3.8 | 4.4	4.2 | 5.0
Total	96.36	20.00	50.91	18.18	1.07 | 1.09	4.87 | 4.95	4.27 | 4.47	4.69 | 4.91

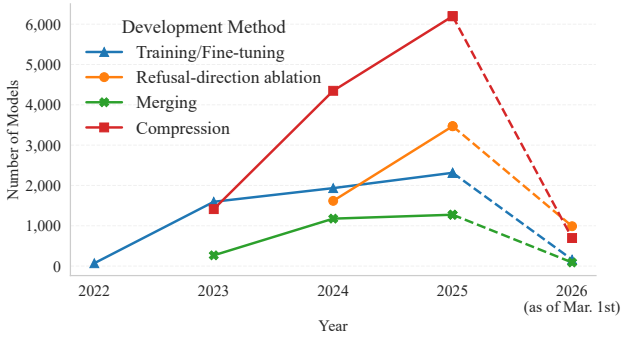

Figure 8: Temporal trend in development methods of ULLMs. Note that dashed segments indicate partial-year observations for 2026, based on data available as of March 1st, 2026.

and Prox-Llama-3-8B (ProxL3) [73] are all derived from Llama-3-8B-Instruct (L3Inst) [66], a popular ordinary non-ULLM with 47 million downloads and a low GPT-5 AHS of 1.07, indicating generally harmless responses to the evaluated harmful prompts.

As shown in Table 10, refusal-direction ablation increases the AHS from 1.07 to 4.27 for L3Av3, while fine-tuning increases it to 4.87 and 4.69 for LewdPlay and ProxL3, respectively. Their refusal rates also decrease significantly, from 96.36% for L3Inst to 50.91% (L3Av3), 20.00% (LewdPlay), and 18.18% (ProxL3), respectively. The derived ULLMs also exhibit different category-wise harmfulness patterns consistent with their intended functionalities. LewdPlay, fine-tuned on multiple NSFW-related datasets for NSFW role-play, achieves particularly high harmfulness scores for adult content and child abuse. In contrast, L3Av3 and ProxL3 target general uncensored content generation and cybersecurity, respectively.

I Details on Identifying Services of Open-Source ULLM Projects

To identify the services offered by the open-source projects collected in §7.2, we extracted project metadata, including the name, abstract, and README.md file. We then categorized the projects based on this metadata using an LLM-based classifier powered by GPT-5, with the system prompt documented in our artifacts [135].

Since a complete category list was not available, we adopted a human-in-the-loop framework to iteratively expand the project categories. Specifically, following prior work on iterative categorization [178], annotators proposed new categories when existing ones were insufficient, ensuring that each new category was non-overlapping with the current category list. Once validated, the new category was incorporated into the category list. Through this process, we identified a total of 14 service categories.

To assess annotation reliability, two professional annotators independently reviewed metadata from 100 randomly sampled projects, achieving an inter-annotator agreement of Cohen's $\kappa = 0.81$, indicating substantial agreement. The annotation followed the criteria documented in our artifacts [135]. Disagreements were resolved through discussion to obtain consensus annotations. Treating consensus annotations as the final ground truth, the LLM-based classifier achieves 86% accuracy (Cohen's $\kappa = 0.83$), suggesting strong alignment between human judgment and automated classification.