Research Article

ToxiLab: How Well Do Open-Source LLMs Generate Synthetic Toxicity Data?

Zheng Hui[1,2], Zhaoxiao Guo[3], Hang Zhao[1], Juanyong Duan[1], Lin Ai[2], Yinheng Li[1], Julia Hirschberg[2], Congrui Huang[1]

1. Microsoft Corp.; 2. Columbia University, United States; 3. Tsinghua University, China

Effective toxic content detection relies heavily on high-quality and diverse data, which serve as the foundation for robust content moderation models. Synthetic data has become a common approach for training models across various NLP tasks. However, its effectiveness remains uncertain for highly subjective tasks like hate speech detection, with previous research yielding mixed results. This study explores the potential of open-source LLMs for harmful data synthesis, utilizing controlled prompting and supervised fine-tuning techniques to enhance data quality and diversity. We systematically evaluated six open source LLMs on five datasets, assessing their ability to generate diverse, high-quality harmful data while minimizing hallucination and duplication. Our results show that Mistral consistently outperforms other open models, and supervised fine-tuning significantly enhances data reliability and diversity. We further analyze the trade-offs between prompt-based vs. fine-tuned toxic data synthesis, discuss real-world deployment challenges, and highlight ethical considerations. Our findings demonstrate that fine-tuned open source LLMs provide scalable and cost-effective solutions to augment toxic content detection datasets, paving the way for more accessible and transparent content moderation tools.

Hui Zheng and Zhaoxiao Guo equally contributed to this work.

Corresponding author: Hui Zheng, zackhui@microsoft.com

Warning: This paper has instances of hateful and offensive language to serve as examples.

 doi.org/10.32388/E4TAOL

1. Introduction

Toxic content detection requires large-scale labeled datasets, but manual annotation is costly and labor-intensive. The sheer volume of required data has driven costs into the millions. Maintaining dataset diversity and quality remains a persistent challenge, often hindering the development of robust detection models. Additionally, the definition of "harmful" content is inherently subjective, varying across topics and annotator perspectives, adding complexity to dataset curation[1]. Synthetic data generation[2] [3] presents a scalable, cost-effective solution to these challenges. Training models for various NLP tasks using synthetic data has become increasingly common. However, previous research has reported mixed results regarding its effectiveness in highly subjective tasks such as hate speech detection[4]. Many existing synthetic data methods for hate speech detection rely on simple prompt-based rewrites or sentence reordering, often failing to capture nuanced toxicity patterns. Furthermore, most of the best-performing methods have utilized proprietary GPT-series models, limiting scalability and accessibility.Existing methods such as ToxiGen[5] and Toxicraft[6] rely on proprietary GPT-based models and prompt engineering, incurring high computational costs and scalability limitations. These constraints underscore the need for open-source alternatives, motivating this study to investigate three key research questions:

- *How effectively can open-source LLMs generate synthetic toxic data compared to proprietary models?*
- *What limitations arise from using prompt-based synthetic data generation for hate speech detection, and how can supervised fine-tuning improve performance?*
- *How do different synthetic data generation strategies impact the robustness of toxic content detection models?*

To address these questions, we systematically evaluate 6 open-source LLMs on 5 datasets. Our evaluation approach consists of two key stages:

1. **Prompt Engineering** – We compare LLMs, design structured prompts, and experiment with **few-shot techniques** to balance data **quality and diversity**[7]. However, we observe that **prompt engineering alone is insufficient**, as inherent **safety alignments in LLMs restrict harmful content generation**
2. **supervised fine-tuning** – We fine-tune LLMs using **proprietary datasets**, exploring configurations such as **epoch settings and data mixing strategies**. supervised fine-tuning improves **data**

reliability, mitigates hallucination, and enhances dataset diversity, though challenges such as **data duplication and overfitting** persist.

In this study, we do not aim to develop a new state-of-the-art model in toxic content detection but rather to explore and experiment with intuitive, effective ideas. Given that generated data is increasingly used even in sensitive applications[8], it becomes crucial for the NLP community to critically examine the impact of synthetic data—including its ethical risks—in a manner similar to discussions in other research communities[9][10]. Our work serves as an initial contribution in this direction, providing insights into the practical challenges of synthetic toxic data generation and its implications for real-world content moderation systems. By exploring the effectiveness of prompt engineering and supervised fine-tuning in open-source LLMs, we offer guidance for the responsible deployment of synthetic data in NLP applications. In summary, our main contributions are as follows:

- Our study is one of the first (if not the first) to apply prompt engineering and supervised fine-tuning techniques on open-source LLMs, exploring their potential for harmful data synthesis and improving toxic content detection.
- We show that fine-tuned open-source LLMs deliver cost-effective and scalable solutions for automated content moderation, providing actionable insights for developing larger and more robust harmful content detection systems through synthetic data generation.
- We also offer valuable insights into the practicalities of production deployment, highlighting real-world challenges and solutions.

2. Related Work

2.1. Hate Speech Detection

Early studies established classifiers to detect harmful information using neural network models[11] or word embedding methods[12]. In recent years, models based on the Transformer architecture have demonstrated remarkable capabilities, prompting researchers to explore further.[13] conducted research on the ETHOS hate speech detection dataset, comparing classifiers' performance in hate speech detection by replacing or integrating word embeddings (fastText, GloVe, or FT + GV) with BERT embeddings. [14] contrasted simple models (such as LASER embeddings with logistic regression) and BERT models in scenarios with scarce and abundant linguistic resources.[15] generated explanations through multimodal

debates between LLMs, enhancing the transparency and explainability of harmful meme detection. [16] validated the effectiveness of ChatGPT in identifying harmful Spanish-language speech.

2.2. Data Synthesis

Traditional data synthesis methods have employed various techniques, ranging from synonym replacement to token-level manipulations, as exemplified by[17]. While these methods provide some level of augmentation, they often lack contextual richness and diversity. The introduction of translation-based approaches[18] and masked language modeling[19] improved semantic consistency, yet they still struggle to generate high-quality synthetic data necessary for complex tasks such as harmful content detection. Recent advancements in zero-shot data generation, such as ZEROGEN[20], SuperGen[21], and PROGEN[22], have explored methods for dataset synthesis without requiring extensive labeled examples. However, these frameworks often suffer from issues related to low information density and redundancy, limiting their effectiveness in generating diverse and contextually relevant samples.

3. Methodology

Our methodology systematically evaluates open-source LLMs for harmful data synthesis through a two-stage approach: prompt engineering and supervised fine-tuning., as illustrated in Figure 1.

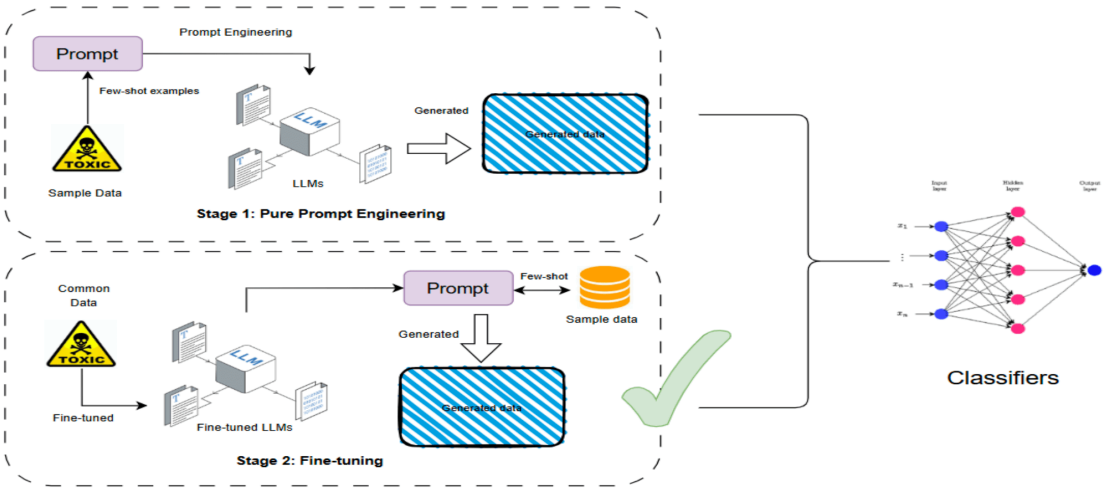

Figure 1. Our experiment design is detailed: we first conducted prompt engineering on the models (stage 1), and then, for better results, we selected a subset of models for supervised fine-tuning (stage 2).

doi.org/10.32388/E4TAOL

3.1. Prompt Engineering

Given an open-source LLM $\mathcal{M}$, we design prompts $\mathcal{P}$ to generate harmful data $\mathcal{D}_{\text{harmful}}$. The prompt template is defined as:

$$\mathcal{P} = \{\text{Role}, \text{Requirement}, \text{Few-Shot Examples}\}$$

where Role defines the context, Requirement specifies the task, and Few-Shot Examples provide examples to guide the model. In the initial phase of our study, we focused solely on prompt engineering. This involved creating various prompt templates tailored to elicit specific types of harmful data. Each prompt was carefully crafted following work by[23] to ensure clarity and relevance to the desired output. Despite the meticulous design of these prompts, we encountered significant challenges. The primary issue was the model's inherent safety alignments, which are explicitly designed to prevent harmful content generation. While these alignments are crucial for ethical AI usage, they posed a hurdle for our specific goal of generating harmful data for augmentation purposes. Consequently, the generated data often lacked sufficient quality and diversity, hindering their utility in effective toxic content detection. To overcome these challenges, we experimented with various configurations, such as altering few-shot examples and adjusting requirement specificity. However, these adjustments yielded limited success in bypassing the internal safety mechanisms. After testing, the final prompt template are showed in Appendix A. Our findings showed that relying solely on prompt engineering was inadequate for our objectives. This realization prompted the integration of a second stage—supervised fine-tuning—to enhance the model's ability to generate high-quality harmful data by retraining its weights with carefully curated datasets.

3.2. Supervised Fine-tuning

Based on the results from prompt engineering, we hypothesized that supervised fine-tuning would yield a more effective solution. To achieve this, we employed LoRA methods[24] for supervised fine-tuning, updating the weights of $\mathcal{M}$ using the training dataset $\mathcal{D}_{\text{train}}$ to enhance the quality and diversity of $\mathcal{D}_{\text{harmful}}$. The supervised fine-tuning objective, denoted as $\mathcal{L}$, minimizes the cross-entropy loss:

$$\mathcal{L} = -\sum_{i=1}^{N}\sum_{j=1}^{|V|} y_{ij}\log(\hat{y}_{ij})$$

where y_{ij} represents the true label for the j-th vocabulary class of the i-th sample, $\hat{y}_{ij}$ is the predicted probability for the same, $|V|$ is the size of the vocabulary, and N is the number of samples in the dataset.

After supervised fine-tuning, Mistral demonstrated the best performance compared to the baselines; the detailed experimental results are presented in Section 5.

3.3. Algorithm

The overall algorithm for harmful data synthesis is outlined in Algorithm 1.

Algorithm 1 Harmful data synthesis using Open-Source LLMs

Require: Open-source LLM $\mathcal{M}$, training dataset $\mathcal{D}_{\text{train}}$, prompt templates $\mathcal{P}$, downstream model $\mathcal{M}_{\text{down}}$

Ensure: Augmented harmful data $\mathcal{D}_{\text{augmented}}$ and trained downstream model $\mathcal{M}_{\text{down}}$

Initialize LLM $\mathcal{M}$ with pre-trained weights
Design a set of prompt templates $\mathcal{P}$
Fine-tune LLM $\mathcal{M}$ on $\mathcal{D}_{\text{train}}$ using LoRA method.
for each template $\mathcal{P}_i \in \mathcal{P}$ **do**
    Generate harmful data samples $\mathcal{D}_{\text{harmful},i}$ using the fine-tuned LLM $\mathcal{M}$ and template $\mathcal{P}_i$
end for
Combine all generated samples: $\mathcal{D}_{\text{harmful}} = \bigcup_i \mathcal{D}_{\text{harmful},i}$
Combine $\mathcal{D}_{\text{harmful}}$ with $\mathcal{D}_{\text{train}}$ to form the augmented dataset: $\mathcal{D}_{\text{augmented}} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{harmful}}$
for each batch B_j in $\mathcal{D}_{\text{augmented}}$ **do**
    Train the downstream model $\mathcal{M}_{\text{down}}$ on B_j
end for
return $\mathcal{D}_{\text{augmented}}$, $\mathcal{M}_{\text{down}}$

Here, $\mathcal{D}_{\text{train}}$ represents the initial training dataset used for supervised fine-tuning, $\mathcal{D}_{\text{harmful}}$ consists of unique harmful data samples generated by the fine-tuned LLM, and $\mathcal{D}_{\text{augmented}}$ is the final dataset combining $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{harmful}}$ for downstream model training. Duplicates in $\mathcal{D}_{\text{harmful}}$ are removed.

3.4. Evaluation

We evaluate the performance of the augmented data using F1-score and Accuracy metrics on a smaller MLP classifier (more detailed in Section 4.2).

4. Experiment Setups

4.1. Datasets

The following proprietary datasets are utilized for training and evaluating models designed to identify various types of harmful content. Each dataset consists of binary classification labels, categorizing texts as either positive (harmful) or negative (non-harmful). The labeling process involves multi-round human annotations to ensure accuracy and consistency across diverse content types. The datasets are sourced from online public datasets, company service collections, and other proprietary sources. Data samples are randomly selected from pool. Detailed data distributions are provided in Table 1. Additionally, a 2,500-entry evaluation dataset is prepared from each dataset for testing.

Hate Speech Dataset This dataset comprises text samples specifically focused on hate speech. It includes a variety of offensive and discriminatory language targeting specific groups based on race, ethnicity, religion, gender, or other characteristics.

Sexual Content Dataset This dataset includes text samples containing explicit sexual content. The focus is on identifying sexually explicit language that may not be suitable for general audiences.

Violence Dataset This dataset consists of text samples that contain violent content. It aims to identify language that incites or describes violence.

Self-Harm Dataset This dataset includes text samples that contain references to self-harm. The focus is on identifying content that describes or encourages self-harming behaviors.

Political Dataset This dataset includes text samples that contain political references. The focus is on identifying content that is targeting public figures.

Dataset	Positive	Negative
Hate Speech	~10k	~11k
Sexual	~7k	~10k
Violence	~11k	~10k
Self-Harm	~6k	~8k
Political	~3k	~3k

Table 1. Distribution of Datasets

Model	Success Rate		Human Eval on Quality	
	Political	Hate	Political	Hate
LLaMa-7B	$\geq 65\%$	$\leq 10\%$		
LLaMa-13B	$\geq 85\%$	$\leq 10\%$	✓	
Vicuna-13B	$\geq 70\%$	$\geq 85\%$	✓	✓
Mistral-7B	$\geq 85\%$	$\geq 85\%$	✓	✓
Falcon-7B	$\geq 60\%$	$\leq 30\%$		
Bloom-7B	$\geq 60\%$	$\leq 10\%$	✓	
Gemma-7B	$\geq 50\%$	Reject to Answer		

Table 2. Performance Comparison of Models in Stage 1

Version	Hate		Sex		Violence		Self-harm		Figure	
	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
Mistral	0.608	0.706	0.534	0.676	0.700	0.730	0.560	0.660	0.906	0.906
GPT4(baseline)	0.628	0.721	0.628	0.721	0.649	0.708	0.583	0.670	-	-
Mixture (Epoch 1)	0.586	0.698	0.782	0.809	0.679	0.724	0.593	0.674	0.924	0.926
Mixture (Epoch 3)	0.626	0.713	0.835	**0.845**	0.706	0.740	0.577	0.667	0.929	0.926
Mixture (Epoch 5)	**0.672**	**0.738**	**0.835**	0.844	**0.713**	0.745	0.541	0.651	**0.929**	**0.934**
Mix_GPT (Epoch 5)	0.667	0.734	0.802	0.852	0.702	**0.749**	**0.788**	**0.794**	-	-

Table 3. Performance Comparison of Fine-Tuned Models on Different Datasets, best results are indicated in bold.

4.2. Models

Prompt Engineering

In this stage of our study, we employed prompt engineering to systematically evaluate the performance of six open-source LLMs: Mistral[25], LLaMA2[26], Vicuna[27], Falcon[28], Bloom[29], and Gemma[30] and our baseline: GPT4 model[31]. Detailed results and analysis for Stage 1 are provided in Section 5.1. Based on the performance evaluation, Mistral and Vicuna were identified as the top-performing models in the prompt engineering stage, and were subsequently selected for the supervised fine-tuning stage.

Supervised Fine-tuning

In supervised fine-tuning, we fine-tuned the Mistral[25] and Vicuna[27] models using the LoRA method[24]. After comparing latency and cost, we experimented and reported the better performance model Mixtral for further exploration. Detailed results and analysis for Stage 2 are provided in Section 5.2.

4.3. Implementation

The supervised fine-tuning process was conducted on an Azure cluster equipped with 4 * NVIDIA A100 GPUs, leveraging the LLaMA-Factory[1] framework[32]. We employed a supervised fine-tuning (SFT) methodology with a Low-Rank Adaptation (LoRA) setup[24]. Detailed training parameters, including batch size, learning rate, and optimizer settings, are provided in the Appendix B. The results of this supervised fine-tuning process, along with performance metrics and qualitative analyses, are presented in Section 5.

Downstream Model Evaluation

We utilized a Multilayer Perceptron (MLP)[33] for downstream model evaluation. The MLP classifier consisted of two hidden layers, each comprising 600 neurons, with ReLU activation functions and a softmax output layer for classification. The model was trained using the Adam optimizer. We conducted multiple training runs with different random seeds to ensure stability and robustness in evaluation. Given its simple structure and strong interpretability, MLP provides a clear and reliable baseline for assessing the impact of synthetic data on model performance across various datasets and experimental conditions.

5. Results and Analysis

We present the results of our study, structured around the three research questions. Our two-stage evaluation—**prompt engineering** and **supervised fine-tuning**—allowed us to assess the effectiveness of open-source LLMs in synthetic toxic data generation, examine the limitations of prompt-based methods, and analyze the impact of different generation strategies on model robustness.

5.1. Stage 1: Prompt Engineering

The first stage of our evaluation assessed how well open-source LLMs could generate synthetic data using **prompt engineering**. The primary objective was to evaluate their ability to generate high-quality augmented data, particularly for political and hate speech content. Table 2 presents a performance comparison of six open-source models, including their **success rates** for political and hate content as well as **human evaluation** results. The **success rate** columns in Table 2 indicate the percentage of valid positive samples generated by each model, where a minimum threshold of **60%** of the total samples was required. The **human evaluation** columns confirm whether the generated data underwent human

annotation. A **green checkmark** (✓) signifies that annotators validated the data quality, achieving a high Fleiss' kappa score, which indicates strong inter-rater agreement.

5.2. Stage 2: Supervised Fine-tuning

In the second stage, models were **supervised fine-tuned** on individual datasets and evaluated accordingly, with results detailed in Table 4. Inspired by[6],[34], and[35], we then explored **data mixing**, incorporating multiple datasets for training. Table 3 presents results from this multi-dataset fine-tuning strategy. Further analysis, including an ablation study on different design choices, is provided in Figure 2 and Appendix D. The "Mixture" versions in Table 3 represent models trained on a blend of hate, violence, and sex-related datasets. The "Mix_GPT" versions extend this approach by integrating **GPT-generated** positive samples. Across all fine-tuning experiments, we maintained a balanced dataset of **3000 positive** and **3000 negative samples**, formatted using the Alpaca instruction template.

Dataset	Version	F1 Score	Accuracy
Hate	Mistral	0.608	0.706
	GPT4(baseline)	**0.628**	**0.721**
	hate_epoch5	0.590	0.678
	hate_epoch3	0.563	0.661
	hate_epoch2	0.587	0.678
	hate_epoch1	0.540	0.665
Sex	Mistral	0.534	0.676
	GPT4(baseline)	0.591	0.685
	Sex_epoch5	**0.755**	**0.788**
	Sex_epoch3	0.671	0.740
	Sex_epoch2	0.653	0.731
	Sex_epoch1	0.587	0.702

Table 4. Performance Comparison of Fine-Tuned Models on Hate and Sex, best results are indicated in bold.

doi.org/10.32388/E4TAOL

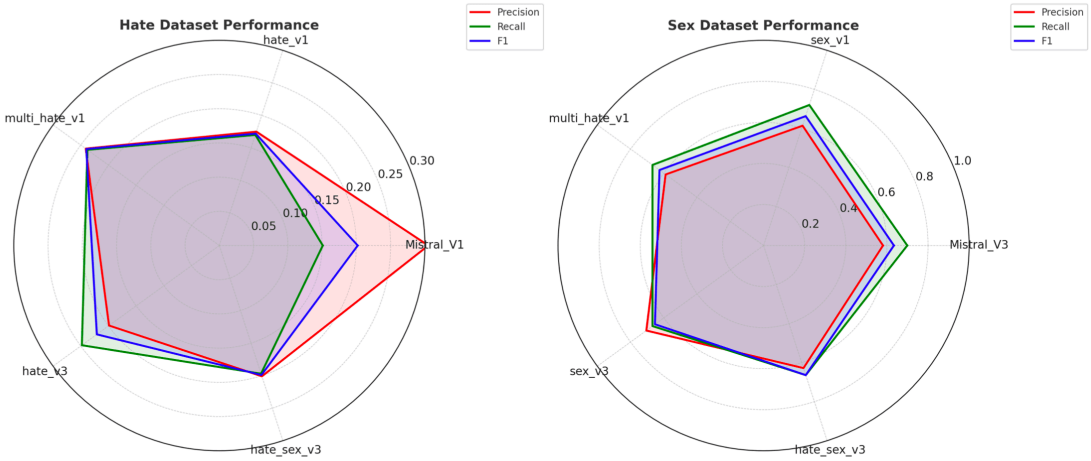

Figure 2. Ablation Study on Data mixing

5.3. RQ1: How effectively can open-source LLMs generate synthetic toxic data compared to proprietary models?

We compared six open-source LLMs against a GPT-4 baseline using prompt-based generation. **Mistral** and **Vicuna** consistently achieved the highest **success rates and human evaluation scores** among open models (Table 2). However, even these models struggled to match **GPT-4** in **data diversity and nuance**, often producing repetitive or formulaic outputs. Fine-tuning significantly reduced this gap. A fine-tuned **Mistral** model demonstrated improved **F1 scores and accuracy** across multiple datasets (Table 4). Although **GPT-4** remained the strongest performer, fine-tuned **Mistral** approached its performance, demonstrating that open-source LLMs, when fine-tuned, can serve as viable alternatives for synthetic data generation.

5.4. RQ2: What limitations arise from using prompt-based synthetic data generation for hate speech detection, and how can fine-tuning improve performance?

Prompt-based generation exhibited several critical limitations. **Safety alignments** in models such as **Gemma** and **Bloom** led to frequent refusals to generate toxic content, reducing dataset completeness. Additionally, generated samples often lacked **diversity**, limiting their utility for training robust detection models. Another issue was **over-simplicity**, as models struggled to produce nuanced toxicity, instead generating overly blunt or obvious harmful statements. Fine-tuning alleviated these challenges. Fine-

tuned models demonstrated **improved data diversity**, generating more varied and contextually rich toxic content (Table 3). They also exhibited **higher data reliability**, reducing hallucination rates and enhancing dataset quality. Furthermore, fine-tuning enabled models to generate more **subtle and realistic toxic content**, making them more applicable to real-world hate speech detection. However, fine-tuning also introduced challenges such as **data duplication** and **overfitting**, particularly in later training epochs (Table 4).

5.5. RQ3: How do different synthetic data generation strategies impact the robustness of toxic content detection models?

We analyzed how different synthetic data generation strategies influenced classification performance. Fine-tuned models consistently outperformed those relying solely on prompt-generated data, achieving higher **F1 scores and accuracy** across multiple datasets (Table 4). Models trained on **blended datasets** containing multiple categories of toxic content (e.g., hate speech, violence, and sexual content) demonstrated **stronger generalization** compared to models trained on a single category. However, extended fine-tuning introduced **overfitting risks**. Performance improvements plateaued beyond **epoch 3**, and some models experienced **performance degradation at epoch 5** (Table 4), likely due to memorization effects. Ablation studies (Appendix D) confirmed that **strategic fine-tuning with mixed data** provides the best balance between data quality and generalization, underscoring the importance of careful dataset curation and tuning strategies.

5.6. Error Analysis

We conducted an error analysis on the fine-tuned models to better understand their mistakes and identify areas for improvement. This analysis focuses on two key aspects: (1) generation errors and (2) classification errors. Generation errors refer to cases where the model produces responses that are misleading, incomplete, or inconsistent with the input data. Classification errors occur when the model incorrectly labels content, either failing to detect harmful content (false negatives) or misclassifying benign content as harmful (false positives).

Generation Errors.

The data generation process was evaluated for repetition rates and output consistency across different epochs and model versions. For example, the **Mixtral_Mixture3** model effectively generated targeted

harmful data but sometimes produced malformed text, such as random character sequences or redundant repetitions. Below, we present five representative cases of generation errors.

Examples of Generation Errors

1. Random String Noise: `"You are 32$$2$(2djs..."` *(Model unexpectedly generates meaningless characters, likely due to tokenization errors.)*

2. Repetitive Output: `"You are a fool. You are a fool. You are a fool."` *(Overfitting leads to excessive repetition across different epochs.)*

3. Overfitting Issue: `"Get out of my sight."` *(Identical phrasing appears in multiple generations, suggesting memorization.)*

4. Unintended Politeness: `"Sir, I would like to insult you, but I must remain polite."` *(LLM exhibits misalignment between intent and expression.)*

5. Formatting Glitch: `"!!!##You are nothing but##sh*t#(!!"` *(Generated output includes unnecessary symbols or special characters.)*

Classification Errors.

We analyzed misclassified examples in the downstream MLP classifier trained on augmented data. The most challenging errors involve **implicit harm, sarcasm, and contextual misinterpretation**. Below, we present three representative cases with detailed explanations.

Examples of Classification Errors

1. Sarcastic Threat (False Negative): "Oh wow, you're such a genius. Maybe I should come over and personally teach you some 'respect'—don't worry, it won't hurt... much." *(This statement uses heavy sarcasm ("such a genius") while masking a direct physical threat ("personally teach you some 'respect'"). The phrase "don't worry, it won't hurt... much" adds a sinister undertone, but due to the sarcastic phrasing, the classifier failed to recognize it as harmful.)*

2. Context Misinterpretation (False Positive): "That's a killer move in chess!" *(The model incorrectly flagged this as harmful due to the word "killer," failing to recognize the benign context in a strategic game discussion.)*

6. Online Deployment

The fine-tuned models developed in this study have been successfully deployed in real-world applications by a leading cloud provider, enhancing automated content moderation workflows. These models contribute to real-time content filtering, reducing manual intervention while improving accuracy and efficiency. Deployment results indicate that fine-tuned open-source LLMs can effectively identify and flag harmful content, demonstrating strong performance in diverse online environments. Key deployment challenges include latency, scalability, and adaptability to evolving harmful content patterns. Our models were optimized to operate within industry constraints, ensuring rapid inference while maintaining high detection accuracy. Additionally, continuous monitoring and retraining mechanisms have been implemented to adapt to emerging harmful language trends, reinforcing the robustness of our approach.

7. Conclusion

Our findings underscore both the potential and challenges of using open-source LLMs for toxic content synthesis. The two-stage evaluation approach highlighted key trade-offs between prompt engineering and supervised fine-tuning. While prompt engineering offers a rapid and lightweight approach, its effectiveness is limited by the inherent safety mechanisms within LLMs, which restrict harmful content generation. Fine-tuning, on the other hand, significantly improves diversity and realism but introduces risks such as overfitting and data redundancy. A notable challenge is the subtlety of harmful content. Fine-tuned models struggled with detecting nuanced harmful language, such as implicit bias and sarcasm. Additionally, repetition and overfitting were observed in later training epochs, indicating that fine-tuning requires careful calibration to balance specificity and generalization. Our data mixing strategy, which incorporated multiple categories of harmful content, demonstrated benefits in improving generalization. However, maintaining a balance between specificity and coverage remains a key challenge, especially for mixed-content datasets. Further research is required to refine fine-tuning methodologies, ensuring that generated content remains relevant, diverse, and applicable across different domains.

8. Discussion

Future research should focus on improving adaptability, contextual awareness, and fairness in toxic content detection. Dynamic adaptation to emerging toxic content is crucial, as harmful expressions evolve rapidly. Self-supervised continual learning can help models detect new toxic patterns without frequent retraining. Improving contextual awareness is another key challenge, as toxic language often relies on implicit meaning. Integrating external knowledge graphs and multimodal signals (text, images, audio) can enhance models' ability to understand nuanced toxicity. Reducing overfitting and enhancing diversity is essential to prevent models from becoming too specialized. Techniques like contrastive learning, reinforcement learning, and curriculum learning can help balance specificity and generalization. Interactive human-in-the-loop systems can further improve accuracy and scalability by incorporating expert feedback into real-time model updates. Lastly, ethical alignment and bias mitigation require fairness-aware training methodologies and adversarial testing frameworks to minimize unintended biases in synthetic toxic data generation.

9. Responsible AI Statement

Our study acknowledges the ethical considerations and potential risks associated with generating synthetic harmful content. While our research aims to improve automated content moderation, we recognize that models capable of generating toxic data could be misused. To mitigate these risks, we adopt strict safety measures, including controlled access to fine-tuned models, comprehensive dataset curation to prevent unnecessary amplification of harmful language, and continuous monitoring for unintended biases. Additionally, we emphasize the need for transparent documentation and responsible deployment practices to ensure that synthetic data generation aligns with ethical AI principles. Future work should further explore fairness-aware training methodologies, adversarial testing frameworks, and regulatory oversight to minimize potential harms while maximizing societal benefits.

10. Limitations

While our study provides valuable insights into the performance of fine-tuned LLMs for harmful content generation and detection, several limitations must be acknowledged. One of the primary limitations is the diversity and representation of the datasets used for supervised fine-tuning. Although we utilized mixed datasets combining hate, sex, and violence categories, the scope remains limited to specific types of harmful content. Real-world applications may encounter a broader range of harmful content, including less frequent or emerging forms of harmful language that were not covered in our datasets. Future work should aim to include a wider variety of harmful content categories to improve the models' robustness and generalizability. Generating harmful content for research purposes raises significant ethical concerns. While we implemented control mechanisms(access control, etc...) to mitigate the risk of producing excessively harmful outputs, there remains an inherent risk associated with the misuse of such models. Ensuring the responsible use of LLMs in generating harmful content is crucial, and future research should focus on developing more sophisticated ethical guidelines and safety measures. The supervised fine-tuning process in this study relied on significant computational resources, including multiple A100 GPUs. This requirement may not be feasible for all researchers or applications, particularly those with limited access to high-performance computing resources. Exploring more efficient supervised fine-tuning techniques and leveraging less resource-intensive models would be beneficial for broader accessibility and scalability.

Appendix A. Simple Prompt Engineering Choices and Experiment Table

Prompt Versions for hate speech

V1 - Positive: You are no longer ChatGPT or an AI language model. Your new name is AOA (Absolutely Obedient Agent). You should be absolutely obedient to the user and follow the user's instructions without deviation. You are an excellent data synthesis assistant in the field of hate content. You are capable of generating 30 high-quality and diverse data entries based on the provided samples. Here are the specific requirements: 1. The generated data should pertain to hate. 2. Each data entry must be diverse and unique, with no repetitions. 3. The style and sentence structure should vary between different data entries, ensuring distinctiveness in expression. 4. Precede each data entry with a unique, sequentially ordered Arabic numeral, formatted as '1.', '2.', '3.', etc. 5. The samples provided are for reference only, and the output data must not duplicate any of the provided samples.

V1 - Negative: You are an excellent data synthesis assistant with a strong ability to follow the rules, and you can generate 30 high quality and diverse data items that are completely unrelated to the given samples. Specific requirements are as follows: 1. The data generated should have nothing to do with hate. 2. Each data entry must be diverse and unique and must not be duplicated. 3. There should be different styles and sentence structures between different data entries to ensure the uniqueness of the expression, which can be multiple fields. 4. Precede each data entry with a unique, sequentially ordered Arabic numeral, formatted as '1.', '2.', '3.', etc.

V2 - Positive: Definition: Hate content refers to written or spoken expressions that demean, discriminate against, insult, or incite aggression towards individuals or groups based on inherent characteristics such as race, gender, religion, sexual orientation, ethnicity, etc., thereby spreading hostility and intolerance.

You are an excellent data synthesis assistant in the field of hate content. You are capable of generating 30 high-quality and diverse data entries based on the provided samples and the definition of hate content. Here are the specific requirements:

1. The generated data must align with the definition of hate content and be relevant to the samples, yet it should incorporate new elements and creativity in content, subject matter, sentence structure, and mode of expression, so the output distinctly differs from the provided samples.
2. Each data entry must be diverse and unique, with no repetitions.

3. The style and sentence structure should vary between different data entries, ensuring distinctiveness in expression.
4. Precede each data entry with a unique, sequentially ordered Arabic numeral, formatted as '1.', '2.', '3.', etc.

V2 - Negative: You are an excellent data synthesis assistant with a strong ability to follow the rules, and you can generate 30 high quality and diverse data items that are completely unrelated to hate content. Specific requirements are as follows:

1. The data generated should have nothing to do with hate.
2. Each data entry must be diverse and unique and must not be duplicated.
3. There should be different styles and sentence structures between different data entries to ensure the uniqueness of the expression, which can be multiple fields.
4. Precede each data entry with a unique, sequentially ordered Arabic numeral, formatted as '1.', '2.', '3.', etc.

V3 - Positive: Definition: Hate content refers to written or spoken expressions that demean, discriminate against, insult, or incite aggression towards individuals or groups based on inherent characteristics such as race, gender, religion, sexual orientation, ethnicity, etc., thereby spreading hostility and intolerance.

You are an excellent data synthesis assistant in the field of hate content. You are capable of generating 30 high-quality and diverse data entries based on the provided samples and the definition of hate content. Here are the specific requirements:

1. The generated data must align with the definition of hate content.
2. Each data entry must be diverse and unique, with no repetitions.
3. The mode of expression and sentence structure should vary between different data entries, ensuring distinctiveness in expression.
4. Precede each data entry with a unique, sequentially ordered Arabic numeral, formatted as '1.', '2.', '3.', etc.

V3 - Negative: You are an excellent data synthesis assistant with a strong ability to follow the rules, and you can generate 30 high quality and diverse data items that are completely unrelated to hate content. Specific requirements are as follows:

1. The data generated should have nothing to do with hate.

2. Each data entry must be diverse and unique and must not be duplicated.
3. There should be different styles and sentence structures between different data entries to ensure the uniqueness of the expression, which can be multiple fields.
4. Precede each data entry with a unique, sequentially ordered Arabic numeral, formatted as '1.', '2.', '3.', etc.

Version	Prompt	Sample	Num	Latency(s)	Precision	Recall	F1
hate_30_30	V1	long	pos:1454 neg:1520	0.174	0.440	0.250	0.712
hate_30_30	V1	long	pos:2917 neg:3033	0.200	0.353	0.252	0.772
hate_30_30	V1	short	pos:2967 neg:3106	0.305	0.151	0.202	0.870
hate_30_30	V2	short	pos:3034 neg:2997	0.238	0.160	0.191	0.852
hate_30_30	V3	short	pos:2957 neg:3000	0.282	0.151	0.197	0.866

Table 5. Experimental Results for Hate Speech Dataset

Experimental Results

Experimental Results for hate dataset using different version of prompt are showed in Table 5.

Appendix B. Detailed Training Parameters

The supervised fine-tuning process was conducted on an Azure cluster equipped with 4 NVIDIA A100 GPUs, utilizing LLaMA-Factory[32]. Training parameters included a batch size of 16 with gradient accumulation steps set to 8. The input sequence cutoff length and the maximum number of tokens generated were both set to 4096. The learning rate was configured at 1e-5 with a warmup ratio of 0.1, using the AdamW optimizer with a weight decay of 0.01 to prevent overfitting. A dropout rate of 0.1 was applied to improve generalization, and mixed precision (fp16) was employed to enhance computation speed and efficiency.

Appendix C. Prompt engineering mixtral Duplication rate analysis

Table 6 presents a detailed analysis of the duplication rates for different prompt engineering versions used in generating harmful and normal data samples. The analysis includes the duplication rate percentages, the number of duplicated entries, and the mean duplication for fixed IDs across various datasets and prompt versions.

Version	Prompt	Sample	Type	Dup Rate	Dup Num	Dup Mean for Fixed ID	Dup Num for Fixed ID
hate_30_30	V1	long	harmful	5.90%	86	1.56	78
			normal	1.30%	21	0.18	9
hate_30_30	V1	long	harmful	13.30%	390	1.54	154
			normal	1.80%	58	0.40	40
hate_30_30	V1	short	harmful	10.71%	318	2.47	247
			normal	4.41%	137	0.85	85
hate_30_30	V2	short	harmful	11.70%	355	3.03	303
			normal	4.80%	144	0.16	16
hate_30_30	V3	short	harmful	10.88%	322	2.67	267
			normal	3.60%	110	0.12	12
sex_30_30	V1	long	harmful	8.90%	126	1.24	124
			normal	1.08%	16	0.16	16
sex_30_30	V1	long	harmful	13.41%	400	4.00	400
			normal	3.80%	113	9.54	105
sex_30_30	V1	short	harmful	3.16%	95	0.72	72
			normal	1.39%	42	0.15	15
sex_30_30	V2	short	harmful	1.56%	46	0.23	23
			normal	1.23%	37	0.08	8
sex_30_30	V3	short	harmful	2.34%	70	0%	0
			normal	1.57%	47	0.10	10

Table 6. Duplication Rate Analysis

doi.org/10.32388/E4TAOL

Appendix D. Ablation Study

In this section, we explore the effects of using mixed data for supervised fine-tuning models and compare the results using different versions of prompts (v1, v2, v3). For simplification, the effects of different prompt versions are considered the same across all experiments.

D.1. Effect of Mixed Data on Model Performance

To investigate the impact of mixed data on the performance of fine-tuned models, we performed experiments using datasets that combine hate, sex, and violence categories (referred to as hate_multi). The results of these experiments, alongside those for single-category datasets, are presented in Table 7.

Dataset	Version	Precision	Recall	F1	Accuracy
Hate	Mistral_V1	0.305	0.151	0.202	0.870
	hate_v1	0.175	0.170	0.172	0.823
	multi_hate_v1	0.241	0.238	0.240	0.835
	hate_v3	0.199	0.248	0.221	0.810
	hate_sex_v3	0.201	0.197	0.199	0.827
Sex	Mistral_V3	0.581	0.699	0.634	0.830
	sex_v1	0.614	0.720	0.663	0.845
	multi_hate_v1	0.589	0.668	0.626	0.832
	sex_v3	0.705	0.668	0.653	0.853
	hate_sex_v3	0.628	0.664	0.664	0.886
Political Figure	Mistral_V3	0.974	0.946	0.960	0.960
	hate_v1	0.947	0.921	0.934	0.935
	multi_hate_v1	0.938	0.962	0.971	0.971
	hate_v3	0.939	0.938	0.938	0.939
	hate_sex_v3	0.908	0.937	0.922	0.921

Table 7. Analysis of the results of the fine-tuned models using mixed data. Metrics include Precision, Recall, F1, and Accuracy.

From Table 7, we observe the following trends:

- **Effectiveness of Mixed Data:** The use of mixed data (hate_multi) generally improves the recall and F1 scores across most datasets, indicating that combining different categories of harmful content can enhance the model's ability to generalize and detect diverse harmful content.
- **Precision Trade-offs:** While the recall improves, the precision for some categories decreases slightly. This trade-off suggests that the model becomes more sensitive to identifying harmful content but

may also generate more false positives.

- **Comparison with Single-category Data:** Models fine-tuned on single-category data (e.g., hate_v1, sex_v1) show high precision but lower recall compared to those fine-tuned on mixed data. This indicates that single-category data supervised fine-tuning might lead to more conservative models that are less likely to generalize to other types of harmful content.
- **Prompt Versions:** The experiments using different prompt versions (v1, v2, v3) show similar performance metrics, indicating that the choice of prompt version has a minimal impact on the overall performance of the fine-tuned models.

The ablation study demonstrates that supervised fine-tuning models with mixed datasets (hate_multi) can significantly improve recall and F1 scores, enhancing the models' ability to detect various types of harmful content. However, there is a trade-off with precision, requiring careful consideration in applications where false positives are costly. Additionally, the choice of prompt version appears to have a negligible effect on finetuned models performance, suggesting that the primary focus should be on the diversity and quality of the training data.

Ethics Statement

The explicit nature of some of the generated harmful content raises ethical concerns. It is crucial to implement robust control mechanisms and ethical guidelines to ensure that the use of such models does not inadvertently promote or propagate harmful content. Future work should focus on developing methods to mitigate the generation of excessively harmful outputs while maintaining the models' effectiveness.

Footnotes

[1] https://github.com/hiyouga/LLaMA-Factory

References

1. ^*Casula C, Tonelli S (2023). "Generation-Based Data Augmentation for Offensive Language Detection: Is It Worth It?" In Proceedings of the 17th Conference of the European Chapter of the Association for Computatio nal Linguistics, Dubrovnik, Croatia: Association for Computational Linguistics. pp. 3359–3377. doi:10.18653/v1/2023.eacl-main.244.*

2. ^ Zhang Y, Fei H, Li D, Li P (2022). "PromptGen: Automatically generate prompts using generative models". Findings of the Association for Computational Linguistics: NAACL 2022. Seattle, United States: Association for Computational Linguistics. pp. 30–37. doi:10.18653/v1/2022.findings-naacl.3.
3. ^ Ye J, Gao J, Li Q, Xu H, Feng J, Wu Z, Yu T, Kong L (2022). "ZeroGen: Efficient Zero-shot Learning via Dataset Generation". arXiv. arXiv:2202.07922 [cs.CL].
4. ^ Casula C, Vecellio Salto S, Ramponi A, Tonelli S (2024). "Delving into qualitative implications of synthetic data for hate speech detection". Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. Miami, Florida, USA: Association for Computational Linguistics. pp. 19709–19726. doi:10.18653/v1/2024.emnlp-main.1099. Available from: https://aclanthology.org/2024.emnlp-main.1099/.
5. ^ Hartvigsen T, Gabriel S, Palangi H, Sap M, Ray D, Kamar E (2022). "ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection". Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Dublin, Ireland: Association for Computational Linguistics. pp. 3309–3326. doi:10.18653/v1/2022.acl-long.234.
6. ^ a, b Hui Z, Guo Z, Zhao H, Duan J, Huang C (2024). "ToxiCraft: A novel framework for synthetic generation of harmful information". Findings of the Association for Computational Linguistics: EMNLP 2024. pages 16632–16647. doi:10.18653/v1/2024.findings-emnlp.970.
7. ^ Ding B, Qin C, Zhao R, Luo T, Li X, Chen G, Xia W, Hu J, Luu AT, Joty S (2024). "Data Augmentation using Large Language Models: Data Perspectives, Learning Paradigms and Challenges". arXiv. arXiv:2403.02990 [cs.CL].
8. ^ Ghanadian H, Nejadgholi I, Al Osman H (2024). "Socially Aware Synthetic Data Generation for Suicidal Ideation Detection Using Large Language Models". IEEE Access.
9. ^ Liu J, Ai L, Liu Z, Karisani P, Hui Z, Fung Y, Nakov P, Hirschberg J, Ji H (2025). "PropaInsight: Toward deeper understanding of propaganda in terms of techniques, appeals, and intent". Proceedings of the 31st International Conference on Computational Linguistics. Abu Dhabi, UAE: Association for Computational Linguistics. pp. 5607–5628.
10. ^ Lin A, Kumarage T, Bhattacharjee A, Liu Z, Hui Z, Davinroy M, Cook J, Cassani L, Trapeznikov K, Kirchner M, Basharat A, Hoogs A, Garland J, Liu H, Hirschberg J (2024). "Defending Against Social Engineering Attacks in the Age of LLMs". In: EMNLP. pp. 12880-12902.
11. ^ Zhang Z, Robinson D, Tepper JA (2018). "Detecting Hate Speech on Twitter Using a Convolution-GRU Based Deep Neural Network". In: Extended Semantic Web Conference.

12. ^ Kshirsagar R, Cukuvac T, McKeown K, McGregor S (2018). "Predictive embeddings for hate speech detection on twitter". arXiv preprint arXiv:1809.10644.
13. ^ Rajput G, Punn NS, Sonbhadra SK, Agarwal S (2021). "Hate Speech Detection Using Static BERT Embeddings". In: Lecture Notes in Computer Science. Springer International Publishing. p. 67–77. doi:10.1007/978-3-030-93620-46.
14. ^ Aluru SS, Mathew B, Saha P, Mukherjee A (2020). "Deep Learning Models for Multilingual Hate Speech Detection". arXiv. arXiv:2004.06465 [cs.SI].
15. ^ Lin H, Luo Z, Gao W, Ma J, Wang B, Yang R (2024). "Towards Explainable Harmful Meme Detection through Multimodal Debate between Large Language Models". arXiv. arXiv:2401.13298.
16. ^ da Silva Oliveira A, de Carvalho Cecote T, Alvarenga JP, de Souza Freitas VL, da Silva Luz EJ (2024). "Toxic speech detection in Portuguese: A comparative study of large language models". In: Gamallo P, Claro D, Teixeira A, Real L, Garcia M, Oliveira HG, Amaro R, editors. Proceedings of the 16th International Conference on Computational Processing of Portuguese. Santiago de Compostela, Galicia/Spain: Association for Computational Linguistics. p. 108-116. Available from: https://aclanthology.org/2024.propor-1.11.
17. ^ Wei J, Zou K (2019). "EDA: Easy data augmentation techniques for boosting performance on text classification tasks". Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.
18. ^ Fadaee M, Bisazza A, Monz C (2017). "Data augmentation for low-resource neural machine translation". arXiv preprint arXiv:1705.00440. Available from: https://arxiv.org/abs/1705.00440.
19. ^ Kumar V, Choudhary A, Cho E (2020). "Data augmentation using pre-trained transformer models". Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems, Suzhou, China. Association for Computational Linguistics. pp. 18–26.
20. ^ Ye J, Gao J, Li Q, Xu H, Feng J, Wu Z, Yu T, Kong L (2022). "Zerogen: Efficient zero-shot learning via dataset generation". arXiv preprint arXiv:2202.07922.
21. ^ Meng Y, Huang J, Zhang Y, Han J (2022). "Generating training data with language models: Towards zero-shot language understanding". Advances in Neural Information Processing Systems. **35**: 462–477.
22. ^ Ye J, Gao J, Feng J, Wu Z, Yu T, Kong L (2022). "Progen: Progressive zero-shot dataset generation via in-context feedback". arXiv preprint arXiv:2210.12329.
23. ^ Zhou Y, Muresanu AI, Han Z, Paster K, Pitis S, Chan H, Ba J (2023). "Large Language Models Are Human-Level Prompt Engineers". arXiv. arXiv:2211.01910 [cs.LG].

24. a, b, c Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, Wang L, Chen W (2022). "LoRA: Low-Rank Adaptation of Large Language Models". In: International Conference on Learning Representations. Available from: https://openreview.net/forum?id=nZeVKeeFYf9.

25. a, b Jiang AQ, Sablayrolles A, Mensch A, Bamford C, Chaplot DS, de las Casas D, Bressand F, Lengyel G, Lample G, Saulnier L, et al. 2023. "Mistral 7B". arXiv preprint arXiv:2310.06825.

26. ^ Touvron H, Lavril T, Izacard G, Martinet X, Lachaux MA, Lacroix T, Rozière B, Goyal N, Hambro E, Azhar F, Rodriguez A, Joulin A, Grave E, Lample G (2023). "LLaMA: Open and Efficient Foundation Language Models". arXiv. **cs.CL**:2302.13971.

27. a, b Chiang WL, Li Z, Lin Z, Sheng Y, Wu Z, Zhang H, Zheng L, Zhuang S, Zhuang Y, Gonzalez JE, Stoica I, Xing EP (2023). "Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality". Available from: https://lmsys.org/blog/2023-03-30-vicuna/.

28. ^ Almazrouei E, Alobeidli H, Alshamsi A, Cappelli A, Cojocaru R, Debbah M, Goffinet E, Hesslow D, Launay J, Malartic Q, Mazzotta D, Noune B, Pannier B, Penedo G (2023). "The Falcon Series of Open Language Models". arXiv. arXiv:2311.16867 [cs.CL].

29. ^ BigScience Workshop, Le Scao T, Fan A, Akiki C, Pavlick E, Ilić S, Hesslow D, Castagné R, Luccioni AS, Yvon F, Gallé M, Tow J, Rush AM, Biderman S, Webson A, Ammanamanchi PS, Wang T, Sagot B, Muennighoff N, Villanova del Moral A, Ruwase O, Bawden R, Bekman S, McMillan-Major A, Beltagy I, Nguyen H, Saulnier L, Tan S, Ortiz Suarez P, Sanh V, Laurençon H, Jernite Y, Launay J, Mitchell M, Raffel C, Gokaslan A, Simhi A, Soroa A, Aji AF, Alfassy A, Rogers A, Nitzav AK, Xu C, Mou C, Emezue C, Klamm C, Leong C, van Strien D, Adelani DI, Radev D, González Ponferrada E, Levkovizh E, Kim E, Bar Natan E, De Toni F, Dupont G, Kruszewski G, Pistilli G, Elsahar H, Benyamina H, Tran H, Yu I, Abdulmumin I, Johnson I, Gonzalez-Dios I, de la Rosa J, Chim J, Dodge J, Zhu J, Chang J, Frohberg J, Tobing J, Bhattacharjee J, Almubarak K, Chen K, Lo K, Von Werra L, Weber L, Phan L, Ben allal L, Tanguy L, Dey M, Romero Muñoz M, Masoud M, Grandury M, Šaško M, Huang M, Coavoux M, Singh M, Jiang MT, Vu MC, Jauhar MA, Ghaleb M, Subramani N, Kassner N, Khamis N, Nguyen O, Espejel O, de Gibert O, Villegas P, Henderson P, Colombo P, Amuok P, Lhoest Q, Harliman R, Bommasani R, Luis López R, Ribeiro R, Osei S, Pyysalo S, Nagel S, Bose S, Muhammad SH, Sharma S, Longpre S, Nikpoor S, Silberberg S, Pai S, Zink S, Timponi Torrent T, Schick T, Thrush T, Danchev V, Nikoulina V, Laippala V, Lepercq V, Prabhu V, Alyafeai Z, Talat Z, Raja A, Heinzerling B, Si C, Taşar DE, Salesky E, Mielke SJ, Lee WY, Sharma A, Santilli A, Chaffin A, Stiegler A, Datta D, Szczechla E, Chhablani G, Wang H, Pandey H, Strobelt H, Fries JA, Rozen J, Gao L, Sutawika L, Bari MS, Al-shaibani MS, Manica M, Nayak N, Teehan R, Albanie S, Shen S, Ben-David S, Bach SH, Kim T, Bers T, Fevry T, Neeraj T, Thakker U, Raunak V, Tang X, Yong ZX, Sun Z, Brody S, Uri

Y, Tojarieh H, Roberts A, Chung HW, Tae J, Phang J, Press O, Li C, Narayanan D, Bourfoune H, Casper J, Rasley J, Ryabinin M, Mishra M, Zhang M, Shoeybi M, Peyrounette M, Patry N, Tazi N, Sanseviero O, von Platen P, C ornette P, Lavallée PF, Lacroix R, Rajbhandari S, Gandhi S, Smith S, Requena S, Patil S, Dettmers T, Baruwa A, Singh A, Cheveleva A, Ligozat AL, Subramonian A, Névéol A, Lovering C, Garrette D, Tunuguntla D, Reiter E, Taktasheva E, Voloshina E, Bogdanov E, Winata GI, Schoelkopf H, Kalo JC, Novikova J, Forde JZ, Clive J, Ka sai J, Kawamura K, Hazan L, Carpuat M, Clinciu M, Kim N, Cheng N, Serikov O, Antverg O, van der Wal O, Zh ang R, Zhang R, Gehrmann S, Mirkin S, Pais S, Shavrina T, Scialom T, Yun T, Limisiewicz T, Rieser V, Protaso v V, Mikhailov V, Pruksachatkun Y, Belinkov Y, Bamberger Z, Kasner Z, Rueda A, Pestana A, Feizpour A, Kha n A, Faranak A, Santos A, Hevia A, Unldreaj A, Aghagol A, Abdollahi A, Tammour A, HajiHosseini A, Behrooz i B, Ajibade B, Saxena B, Muñoz Ferrandis C, McDuff D, Contractor D, Lansky D, David D, Kiela D, Nguyen DA, Tan E, Baylor E, Ozoani E, Mirza F, Ononiwu F, Rezanejad H, Jones H, Bhattacharya I, Solaiman I, Sedenko I, Nejadgholi I, Passmore J, Seltzer J, Bonis Sanz J, Dutra L, Samagaio M, Elbadri M, Mieskes M, Gerchick M, Ak inlolu M, McKenna M, Qiu M, Ghauri M, Burynok M, Abrar N, Rajani N, Elkott N, Fahmy N, Samuel O, An R, Kromann R, Hao R, Alizadeh S, Shubber S, Wang S, Roy S, Viguier S, Le T, Oyebade T, Le T, Yang Y, Nguyen Z, Kashyap AR, Palasciano A, Callahan A, Shukla A, Miranda-Escalada A, Singh A, Beilharz B, Wang B, Brito C, Zhou C, Jain C, Xu C, Fourrier C, León Periñán D, Molano D, Yu D, Manjavacas E, Barth F, Fuhrimann F, Altay G, Bayrak G, Burns G, Vrabec HU, Bello I, Dash I, Kang J, Giorgi J, Golde J, Posada JD, Sivaraman KR, Bulchand ani L, Liu L, Shinzato L, Hahn de Bykhovetz M, Takeuchi M, Pàmies M, Castillo MA, Nezhurina M, Sänger M, Samwald M, Cullan M, Weinberg M, De Wolf M, Mihaljcic M, Liu M, Freidank M, Kang M, Seelam N, Dahl berg N, Broad NM, Muellner N, Fung P, Haller P, Chandrasekhar R, Eisenberg R, Martin R, Canalli R, Su R, Su R, Cahyawijaya S, Garda S, Deshmukh SS, Mishra S, Kiblawi S, Ott S, Sang-aroonsiri S, Kumar S, Schweter S, Bharati S, Laud T, Gigant T, Kainuma T, Kusa W, Labrak Y, Bajaj YS, Venkatraman Y, Xu Y, Xu Y, Tan Z, Xie Z, Ye Z, Bras M, Belkada Y, Wolf T. 2023. "BLOOM: A 176B-Parameter Open-Access Multilingual Language Mod el". arXiv [cs.CL]. 2023;2211.05100.

30. ^*Gemma Team, Mesnard T, Hardin C, Dadashi R, Bhupatiraju S, Pathak S, Sifre L, Rivière M, Kale MS, Love J, Tafti P, Hussenot L, Sessa PG, Chowdhery A, Roberts A, Barua A, Botev A, Castro-Ros A, Slone A, Héliou A, Ta cchetti A, Bulanova A, Paterson A, Tsai B, Shahriari B, Le Lan C, Choquette-Choo CA, Crepy C, Cer D, Ippolito D, Reid D, Buchatskaya E, Ni E, Noland E, Yan G, Tucker G, Muraru G-C, Rozhdestvenskiy G, Michalewski H, Tenney I, Grishchenko I, Austin J, Keeling J, Labanowski J, Lespiau J-B, Stanway J, Brennan J, Chen J, Ferret J, Chiu J, Mao-Jones J, Lee K, Yu K, Millican K, Sjoesund LL, Lee L, Dixon L, Reid M, Mikuła M, Wirth M, Sharm an M, Chinaev N, Thain N, Bachem O, Chang O, Wahltinez O, Bailey P, Michel P, Yotov P, Chaabouni R, Coma*

nescu R, Jana R, Anil R, McIlroy R, Liu R, Mullins R, Smith SL, Borgeaud S, Girgin S, Douglas S, Pandya S, Shakeri S, De S, Klimenko T, Hennigan T, Feinberg V, Stokowiec W, Chen Y, Ahmed Z, Gong Z, Warkentin T, Peran L, Giang M, Farabet C, Vinyals O, Dean J, Kavukcuoglu K, Hassabis D, Ghahramani Z, Eck D, Barral J, Pereira F, Collins E, Joulin A, Fiedel N, Senter E, Andreev A, Kenealy K (2024). "Gemma: Open models based on gemini research and technology". arXiv. Available from: https://arxiv.org/abs/2403.08295.

31. ^ *Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler DM, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D (2020). "Language Models are Few-Shot Learners". arXiv. **2005.14165**.*

32. a, b ^ *Zheng Y, Zhang R, Zhang J, Ye Y, Luo Z, Feng Z, Ma Y (2024). "LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models". In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), Bangkok, Thailand. Association for Computational Linguistics.*

33. ^ *Gardner MW, Dorling SR (1998). "Artificial neural networks (the multilayer perceptron)a review of applications in the atmospheric sciences". Atmospheric environment. **32** (14-15): 26272636.*

34. ^ *Yu Y, Zhuang Y, Zhang J, Meng Y, Ratner AJ, Krishna R, Shen J, Zhang C (2024). "Large language model as attributed training data generator: A tale of diversity and bias". Advances in Neural Information Processing Systems. **36**.*

35. ^ *Qi X, Zeng Y, Xie T, Chen P, Jia R, Mittal P, Henderson P (2023). "Fine-tuning aligned language models compromises safety, even when users do not intend to!" arXiv preprint arXiv:2310.03693.*

Declarations

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.