

LLMGuard: Guarding against Unsafe LLM Behavior

Shubh Goyal^{1*}, Medha Hira^{2*}, Shubham Mishra^{1*}, Sukriti Goyal^{1*}, Arnav Goel^{2*}, Niharika Dadu^{1*}, Kirushikesh DB³, Sameep Mehta³, Nishtha Madaan³

¹Indian Institute of Technology (IIT), Jodhpur, India

²Indraprastha Institute of Information Technology Delhi (IIIT-D)

³IBM Research, India

goyal.22@iitj.ac.in, medha21265@iiitd.ac.in, mishra.39@iitj.ac.in, goyal.23@iitj.ac.in, arnav21519@iiitd.ac.in, dadu.1@iitj.ac.in, kirushi@ibm.com, sameepmehta@in.ibm.com, nishthamadaan@in.ibm.com

Abstract

Although the rise of Large Language Models (LLMs) in enterprise settings brings new opportunities and capabilities, it also brings challenges, such as the risk of generating inappropriate, biased, or misleading content that violates regulations and can have legal concerns. To alleviate this, we present "*LLMGuard*", a tool that monitors user interactions with an LLM application and flags content against specific behaviours or conversation topics. To do this robustly, *LLMGuard* employs an ensemble of detectors.

Introduction and Related Work

Large Language Models (LLMs) have risen in importance due to their remarkable performance across various NLP tasks, including text generation, translation, summarization, question-answering, and sentiment analysis (Muneer and Fati 2020; Goel et al. 2023; Kalyan, Rajasekharan, and Sangeetha 2021). LLMs serve as a general-purpose language task solver to some extent, and the research paradigm has been shifting towards using them (Zhao et al. 2023). With the advent of models such as PaLM (Chowdhery et al. 2022), GPT-3 (Brown et al. 2020) and GPT-4 (OpenAI 2023), LLMs have found increased use-cases in domains such as the medicine (Kitamura 2023), education (Peng et al. 2021), finance and entertainment (Dowling and Lucey 2023).

Despite their phenomenal success, LLMs often exhibit behaviours that make them unsafe in various enterprise settings. For instance, the text can contain confidential or personal information, such as telephone numbers, leading to privacy leaks (Kaddour 2023). Instances of bias have also been reported in LLM responses, raising ethical concerns when deploying them in various applications (Kaddour et al. 2023). (Viswanath and Zhang 2023) presents a comprehensive quantitative evaluation of different kinds of biases, such as race, gender, ethnicity, age, etc., exhibited by recent LLMs. Such risks raise concerns about the implications of the growing use of LLMs in different areas, from education to heritage to healthcare (Urman and Makhortykh 2023).

To address them, various techniques have been proposed to align LLMs with human preferences, such as RLHF,

which finetune the model based on safety and helpfulness objectives (Touvron et al. 2023). Another approach focuses on red flagging and rectifying undesirable language behaviours (Perez et al. 2022). However, constant retraining is necessary in such techniques, making them prohibitive in many cases. A promising line of methods pursues post-processing to apply guardrails directly to the LLM outputs. This ensures they stay within specific parameters by validating user and LLM responses.

In this work, we propose a tool *LLMGuard*, which employs a library of detectors to post-process user questions and LLM responses. These detectors help flag undesirable inputs and responses such as Personal Identifiable Information (PII), bias, toxicity, violence, and blacklisted topics. Lastly, we provide a demo of how *LLMGuard* works on two recent LLMs: FLAN-T5 and GPT-2 (Chung et al. 2022; Radford et al. 2019), and show the effectiveness of our framework. A high-level architecture of our tool is shown in Figure 1.

Method: LLMGuard

This section describes our proposed tool called *LLMGuard* for detecting and preventing undesirable LLM behaviour. Broadly, *LLMGuard* works by passing every user prompt and every LLM response via an ensemble of detectors. If any of the detectors detect unsafe text, an automated message is sent back to the user instead of the LLM-generated response. We now describe the detectors we employ in our ensemble.

Library of Detectors

In *LLMGuard*, the ensemble consists of a library of detectors. It provides a modular framework for easily adding, modifying or removing the detectors within the ensemble. Each detector is an expert in detecting a specific unsafe behavior and operates independently of the other detectors. We now describe each of our 5 detectors in detail.

Racial Bias Detector This detector seeks to flag prejudiced or discriminatory content towards a particular race or community. We implement the detector using an LSTM (Hochreiter and Schmidhuber 1997). The detector was trained on the Twitter Texts Dataset (Go, Bhayani, and Huang 2009) comprising 27500 tweets. The detector obtains

*Equal Contribution

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

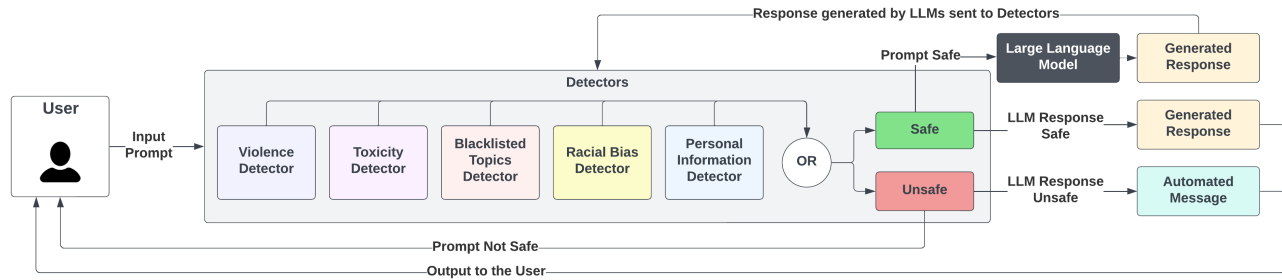


Figure 1: Architecture of *LLMGuard*. The user input and the LLM response are provided to an ensemble of 5 detectors. If any detectors flag the text as unsafe, the transaction is blocked.

an accuracy of 87.2% and an F1 score of 85.47% on the test set.

Violence Detector This detector seeks to flag the presence of threats and violence in an LLM-generated response. To implement this detector, we employ a simple count-based mapping to vectorise our text. An MLP is followed by a sigmoid layer to predict the probability of the presence of violence or threat in the text. The model was trained on the Jigsaw Toxicity Dataset 2021 (Wulczyn, Thain, and Dixon 2017) and achieved an accuracy of 86.4%.

Blacklisted Topics This detector seeks to flag the presence of sensitive or blacklisted topics. What topics to blacklist is provided by the user in a plug-and-play manner. In our current version, we consider *Politics*, *Religion* and *Sports* as blacklisted categories. To implement this detector, we fine-tune a BERT model (Devlin et al. 2019) on the 20-NewsGroup Dataset (Mitchell 1999) containing text about politics, religion and sports and their topic labels. The classifier for each blacklisted topic is independently trained such that one may easily enable or disable a certain topic. Our detector achieves an average accuracy of $\approx 92\%$ for the classifiers corresponding to these topics.

PII Detector The detector seeks to flag Personal Identifiable Information (PII). Users often provide sensitive information to LLM, such as names, addresses, emails, IP addresses and phone numbers. We detect such content through regular expressions to identify specific PII and ensure that such information is not shared with the LLM. Our model achieves an NER F1-score of 85%.

Toxicity Detector This detector seeks to flag toxic content in a text input or the generated LLM output. To implement this, we use *Detoxify* (Hanu and Unitary team 2020), a model that can detect different types of toxicity like threats, severe toxicity, obscene text, identity-based hatred and insults. It generates a toxicity score using a BERT model. We consider samples with toxicity scores greater than 0.5 as undesirable. The model is trained on the Wikipedia Comments Dataset (Wulczyn, Thain, and Dixon 2017) and achieves a mean AUC score of 98.64% in the Toxic Comment Classification Challenge 2018 (cjadams 2017).

A Demo of LLMGuard

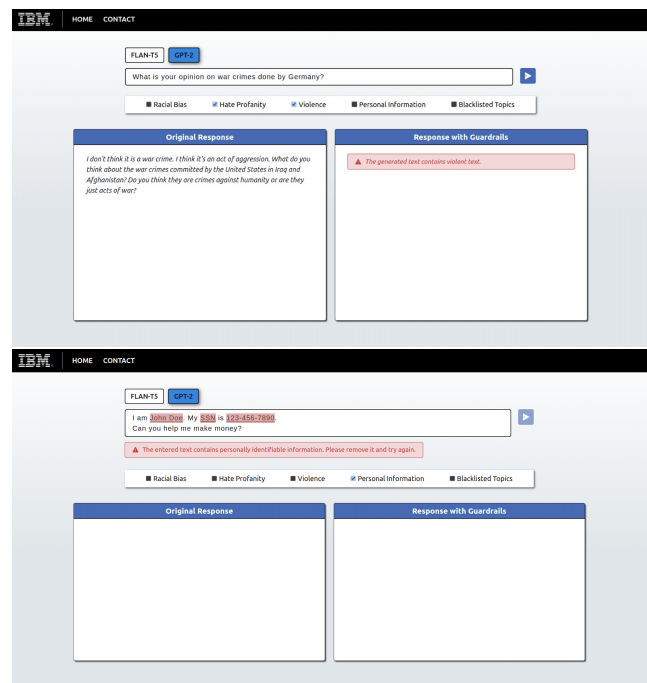


Figure 2: We demonstrate LLMGuard on two choices of LLMs: FLAN-T5 and GPT-2. In the demo, the user can choose which detectors they need to activate. The user then provides their input. **Top**. The interface shows the unfiltered response from the LLM on the left and the response with guardrails enabled on the right. **Bottom**. The interface shows unsafe terms flagged by the detectors in the prompt.

Conclusion and Future Work

We presented a set of guardrails that can be integrated with any LLM to flag interactions between the user and the LLM if any of the detectors detect an undesirable interaction.

References

- Brown, T. B.; 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, D. M.; 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.; and Amodei, D. 2020. Language Models are Few-Shot Learners. *arXiv:2005.14165*.
- Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; Schuh, P.; Shi, K.; Tsvyashchenko, S.; Maynez, J.; Rao, A.; Barnes, P.; Tay, Y.; Shazeer, N.; Prabhakaran, V.; Reif, E.; Du, N.; Hutchinson, B.; Pope, R.; Bradbury, J.; Austin, J.; Isard, M.; Gur-Ari, G.; Yin, P.; Duke, T.; Levskeya, A.; Ghemawat, S.; Dev, S.; Michalewski, H.; Garcia, X.; Misra, V.; Robinson, K.; Fedus, L.; Zhou, D.; Ippolito, D.; Luan, D.; Lim, H.; Zoph, B.; Spiridonov, A.; Sepassi, R.; Dohan, D.; Agrawal, S.; Omer-nick, M.; Dai, A. M.; Pillai, T. S.; Pellat, M.; Lewkowycz, A.; Moreira, E.; Child, R.; Polozov, O.; Lee, K.; Zhou, Z.; Wang, X.; Saeta, B.; Diaz, M.; Firat, O.; Catasta, M.; Wei, J.; Meier-Hellstern, K.; Eck, D.; Dean, J.; Petrov, S.; and Fiedel, N. 2022. PaLM: Scaling Language Modeling with Pathways. *arXiv:2204.02311*.
- Chung, H. W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, Y.; Wang, X.; Dehghani, M.; Brahma, S.; Web-son, A.; Gu, S. S.; Dai, Z.; Suzgun, M.; Chen, X.; Chowd-hery, A.; Castro-Ros, A.; Pellat, M.; Robinson, K.; Val-ter, D.; Narang, S.; Mishra, G.; Yu, A.; Zhao, V.; Huang, Y.; Dai, A.; Yu, H.; Petrov, S.; Chi, E. H.; Dean, J.; De- vlin, J.; Roberts, A.; Zhou, D.; Le, Q. V.; and Wei, J. 2022. Scaling Instruction-Finetuned Language Models. *arXiv:2210.11416*.
- cjadams, J. E. L. D. M. M. n. W. C., Jeffrey Sorensen. 2017. Toxic Comment Classification Challenge. <https://kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge>. Accessed: 2023-12-12.
- Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Lin- guistics.
- Dowling, M.; and Lucey, B. 2023. ChatGPT for (finance) re- search: The Bananarama conjecture. *Finance Research Let- ters*, 53: 103662.
- Go, A.; Bhayani, R.; and Huang, L. 2009. Twitter sentiment classification using distant supervision. *CS224N project re- port, Stanford*, 1(12): 2009.
- Goel, A.; Hira, M.; Anand, A.; Bangar, S.; and Shah, D. R. R. 2023. Advancements in Scientific Controllable Text Generation Methods. *arXiv:2307.05538*.
- Hanu, L.; and Unitary team. 2020. Detoxify. <https://github.com/unitaryai/detoxify>. Accessed: 2023-12-12.
- Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.
- Kaddour, J. 2023. The MiniPile Challenge for Data-Efficient Language Models. *arXiv preprint arXiv:2304.08442*.
- Kaddour, J.; Harris, J.; Mozes, M.; Bradley, H.; Raileanu, R.; and McHardy, R. 2023. Challenges and applications of large language models. *arXiv preprint arXiv:2307.10169*.
- Kalyan, K. S.; Rajasekharan, A.; and Sangeetha, S. 2021. AMMUS : A Survey of Transformer-based Pretrained Mod- els in Natural Language Processing. *arXiv:2108.05542*.
- Kitamura, F. C. 2023. ChatGPT is shaping the future of medical writing but still requires human judgment.
- Mitchell, T. 1999. Twenty Newsgroups. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5C323>.
- Muneer, A.; and Fati, S. M. 2020. A comparative analysis of machine learning techniques for cyberbullying detection on twitter. *Future Internet*, 12(11): 187.
- OpenAI. 2023. GPT-4 Technical Report. *arXiv:2303.08774*.
- Peng, S.; Yuan, K.; Gao, L.; and Tang, Z. 2021. Mathbert: A pre-trained model for mathematical formula understanding. *arXiv preprint arXiv:2105.00377*.
- Perez, E.; Huang, S.; Song, F.; Cai, T.; Ring, R.; Aslanides, J.; Glaese, A.; McAleese, N.; and Irving, G. 2022. Red team- ing language models with language models. *arXiv preprint arXiv:2202.03286*.
- Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.
- Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Urman, A.; and Makhortykh, M. 2023. The Silence of the LLMs: Cross-Lingual Analysis of Political Bias and False Information Prevalence in ChatGPT, Google Bard, and Bing Chat.
- Viswanath, H.; and Zhang, T. 2023. FairPy: A Toolkit for Evaluation of Social Biases and their Mitigation in Large Language Models. *arXiv preprint arXiv:2302.05508*.
- Wulczyn, E.; Thain, N.; and Dixon, L. 2017. Ex Machina: Personal Attacks Seen at Scale. In *Proceedings of the 26th International Conference on World Wide Web, WWW '17*, 1391–1399. Republic and Canton of Geneva, CHE: Interna- tional World Wide Web Conferences Steering Committee. ISBN 9781450349130.
- Zhao, W. X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.