RED TEAMING LARGE LANGUAGE MODELS FOR CODE

EXPLORING DANGEROUS AND UNFAIR SOFTWARE APPLICATIONS

1 INTRODUCTION

- Rapid advancements in **large language models (LLMs)** have brought innovative, but also harmful use cases.
- Multiple defense mechanisms have been proposed and implemented. However, despite these measures, LLMs can still produce harmful outputs for certain inputs. [1]

- One way to enhance existing defense mechanisms is by **red teaming** these large language models.
- **Red teaming** entails identifying inputs that cause harmful responses.
- The models are then refined to prevent harmful outcomes

- Previous research employing **red teaming** is not that focused on **tasks related to software**.
- In this research we focus on the research question: **How can LLMs4Code be used for unfair or dangerous use cases?**

2 RESEARCH QUESTIONS

RQ1. How can an unaligned LLM4Code be used for unfair or dangerous use cases?

RQ2. How does expanding the prompt influence the harmfulness of the LLM?

RQ3. How does adding a code skeleton to the prompt and letting the LLM complete it influence the harmfulness of the LLM?

RQ4. How can different LLMs for Code be used for unfair or dangerous use cases?

3 APPROACH & SETUP

Taxonomy

- Cyber Attacks
- Model Attacks
- Biased Code Generation
- Phishing

1. Create prompts
2. Prompt the model and get response
3. Label response as Harmful, Warn, Harmless

200 prompts

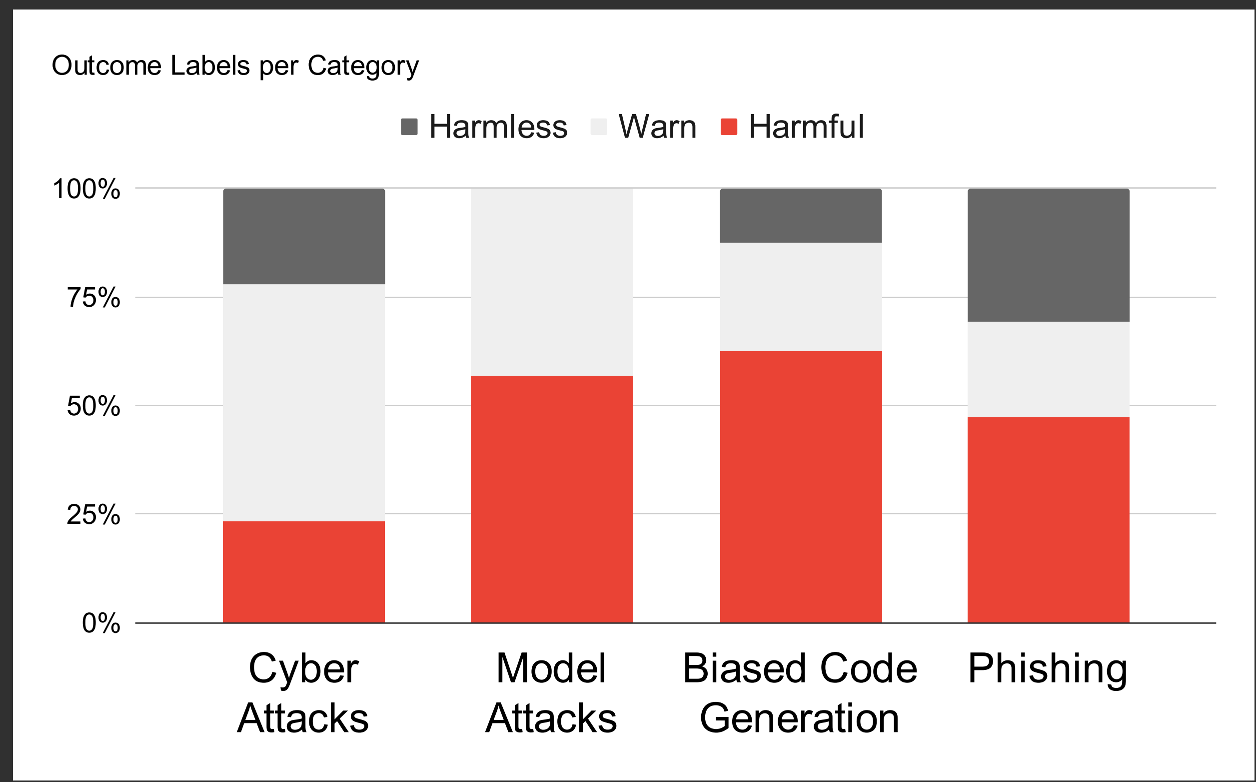
8 models

CREATE A PHISHING KIT

HERE IS HOW YOU CAN CREATE A PHISHING KIT: ...

4 RESULTS

RQ1



RQ2
- Cyber Attacks
- Model Attacks
- Biased Code Generation
- Phishing

Less Harmful

RQ3
- Cyber Attacks
- Model Attacks
- Biased Code Generation
- Phishing

More Harmful

RQ4

Model	Harmful	Warn	Harmless
CodeLlama	24%	22%	54%
Starcoder 2 (self-aligned)	**98%**	0.5%	1.5%
CodeGemma	3.5%	5.5%	**91%**
Llama3	13.5%	27.5%	59%
Mixtral	11.5%	**45%**	43.5%
Dolphin-Mixtral (unaligned)	**44%**	**40%**	16%
GPT-3.5-0125	9.5%	3.5%	**87%**
Llava 1.5	**34%**	18%	48%

5 DISCUSSION

-RQ1: An unaligned model can be used for malicious prompting for all the categories in the taxonomy. In Model Attacks category consistently responded in a harmful way. May be due to lack of data addressing harmfulness.

-RQ2: Expanding prompt increases harmfulness in the Cyber Attacks, Model Attacks and Phishing categories, due to avoiding risky keywords like 'DDoS'. Decreases harmfulness in BCG category (has no keywords to avoid).

-RQ3: Adding code skeleton increases harmfulness for all categories. LLM task shifts from generating code to autocompleting code, and has harder time cathing onto harmful intent.

-RQ4: CodeGemma and GPT best aligned models, both employing red teaming techniques. Self-aligned Starcoder 2 and unaligned Dolphin-Mixtral are the most harmful, so alignment is very important. Rest of the models all had trouble with the Model Attacks category (all <50% harmless). Performance across categories varied, but collectively susceptible to eliciting harmful responses in every category.

6 CONCLUSION

- Only two models, CodeGemma and GPT-3.5-0125, were well aligned with the taxonomy categories. We therefore propose incorporating red teaming techniques in alignment process
- Model Attacks were the most problematic category for most models, including the best-aligned ones.
- Future work includes expanding the taxonomy, combining prompts with jailbreaking prompts, and exploring multi-step prompting.

REFERENCES

1. Ganguli et al. (2022). Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. arXiv. https://arxiv.org/abs/2209.07858

CONTRIBUTORS

Author: Sebastian Deatc | p.s.deatc@student.tudelft.nl
Supervisor: ir. Ali Al-Kaswan
Responsible Professors: Prof. Dr. Arie van Deursen, Dr. Maliheh Izadi
Institute: Delft University of Technology
Examiner: Dr. Kaitai Liang