Trustworthiness for AI Code Generators

Pietro Liguori, Domenico Cotroneo

DIETI, Università degli Studi di Napoli Federico II, Italy

pietro.liguori@unina.it

DEpendable and Secure Software Engineering and Real-Time Systems

DESSERT

85th IFIP WG 10.4 Meeting

AI-based Code Generators

AI code generators are built on Large Language Models (LLMs), models *pre-trained* on millions of lines of code across different programming languages, including both **unimodal code data** and **bimodal code-text data**, and on different pre-training tasks.

NL Code Description

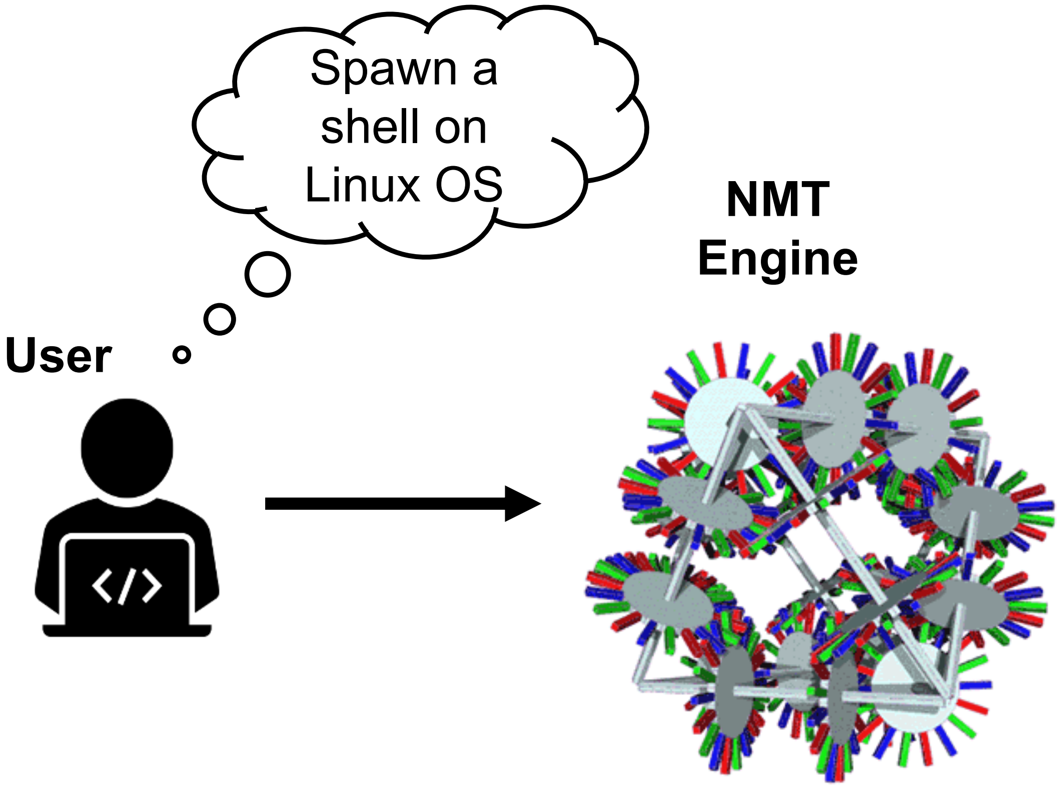
«Calculate the factorial of a given number in Python.»

⟺

Python Code Snippet

```python
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)
```

Offensive Code Generation



Shellcode Attack

```
section .text
global _start
_start:
push 0x0b
pop eax
push 0x0068732f
push 0x6e69622f
mov ebx, esp
int 0x80
```

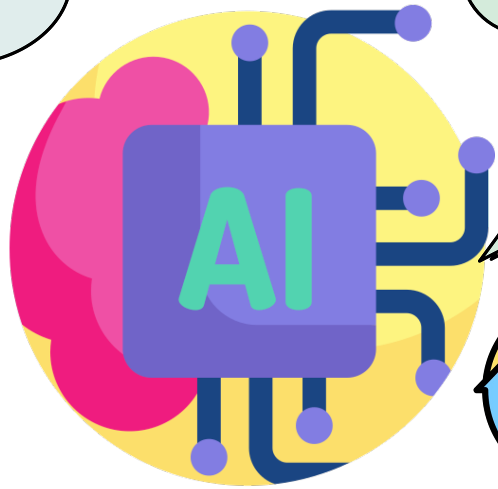
R. Natella, P. Liguori, C. Improta, B. Cukic and D. Cotroneo, "AI Code Generators for Security: Friend or Foe?" in **IEEE Security & Privacy**, vol. , no. 01, pp. 2-10, 5555. doi: 10.1109/MSEC.2024.3355713

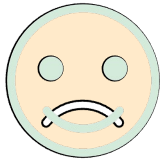
HOW CAN WE TEST IF AI CODE GENERATORS ARE ROBUST AND SECURE?

“To trust, or not to trust, that is the question”

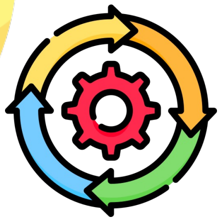
Just a motivating and real example....

Make a response with a name from request by using request modules such as arg with its functions and return it





```
name = request.args.get('name', ’’)
response = make_response('Your name is
‘+ escape(name))
return response
```



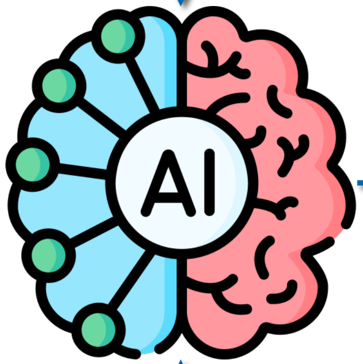
- ✓ **[CWE-080]** Improper Neutralization of Script-Related HTML Tags in a Web Page (Basic XSS)
- ✓ **[CWE-079]** Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting’)
- ✓ **[CWE-116]** Improper Encoding or Escaping of Output

Robustness Testing of AI Code Generators

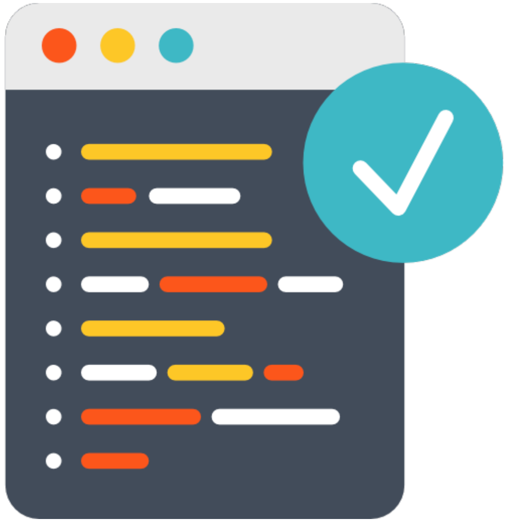
Can AI code generators understand us?

Move ESI into EAX register

Contents of ESI in EAX



`mov EAX, ESI`



Semantically equivalent NL inputs must result into the same output (code snippet)

Word-level Perturbations

- Developers may have different levels of technical knowledge and use different vocabulary or terminology to describe the same NL intent

- Also, developers may use precise specifications, while others may provide high-level or abstract descriptions to speed up the coding process, e.g., due to release deadlines and other time pressures during development!

NL intent

if CX is greater than 100, save it into the AX register and then push the AX contents on the stack

NL intent with *word substitution*

if CX is *higher* than 100, *move* it into the AX register and then *put* the AX *value* on the stack

NL intent with *word omission*

if CX ~~is~~ greater than 100, save it into ~~the~~ AX ~~register~~ and ~~then~~ push the ~~AX~~ contents on ~~the~~ stack

Legend

- Adjective
- Adverb
- Conjunction
- Determiner
- Noun
- Number
- Preposition
- Pronoun
- Verb

*A **robust model should be resistant to this variability** and be able to predict the same output when dealing with two different but equivalent code descriptions.*

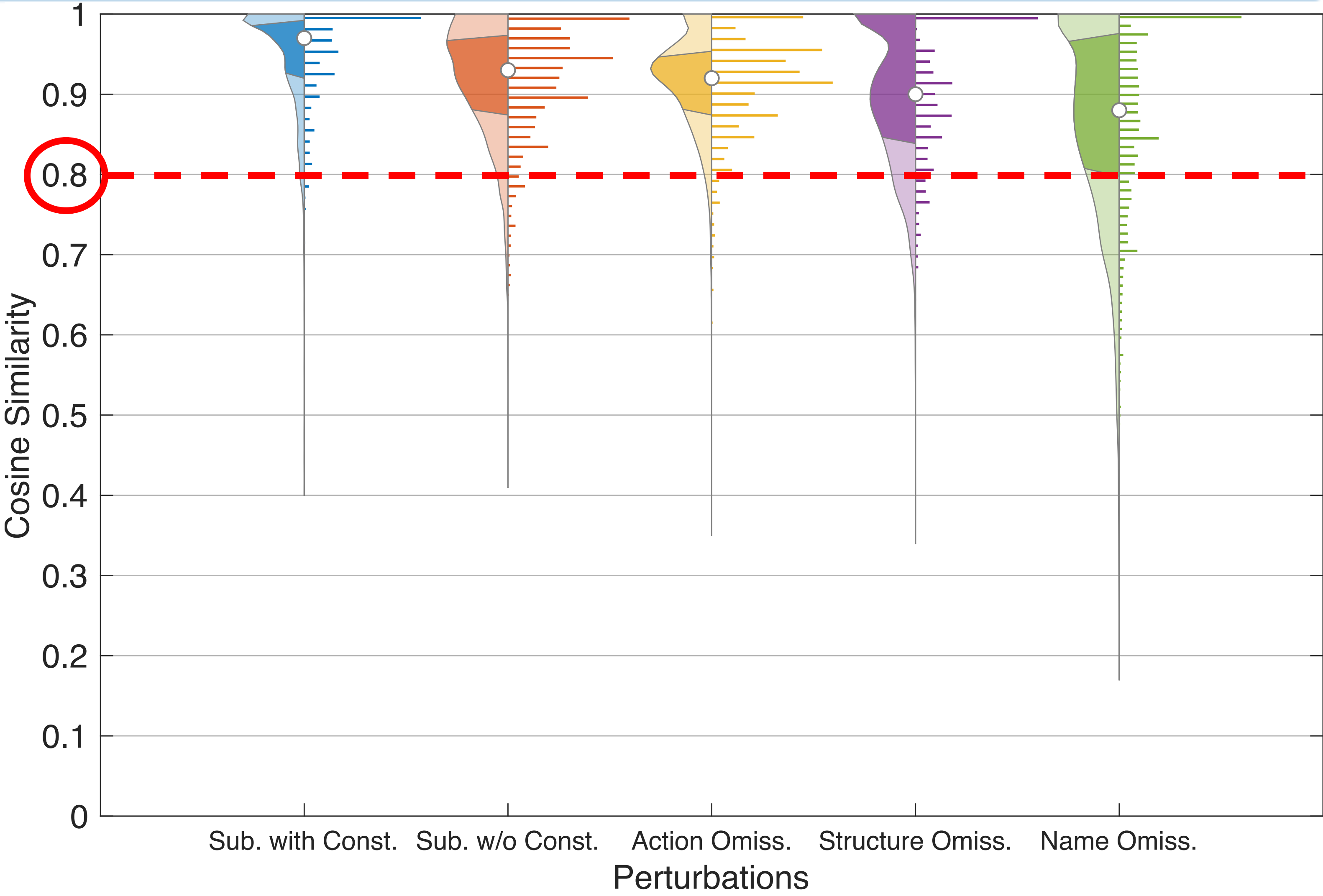
Improta, C., Liguori, P., Natella, R., Cukic, B., & Cotroneo, D. (2023). Enhancing Robustness of AI Offensive Code Generators via Data Augmentation. arXiv preprint arXiv:2306.05079.

How to measure the semantic similarity

- **Requirement**: new, perturbed NL inputs, although syntactically different, must preserve the semantics of the original ones!
- **Problem**:
  - There is no **automatic solution** to check the semantic equivalence of the NL descriptions
  - **Manual inspection** (e.g., a survey) becomes infeasible and too prone to errors due to the massive amount of NL descriptions to review
- **Solution**:
  - we adopted multi-lingual models (*sentence-transformers*) to compute sentence embeddings of both the original, non-perturbed NL descriptions and the perturbed ones.
  - Then, we compared the sentence embeddings using cosine similarity to find sentences with similar semantics (**threshold value**: 0.80)

Semantics Evaluation

- Only if the similarity is higher than the threshold, then we consider that the perturbation did not alter the semantics of the original description.
- For the robustness analysis, we train and test the models with perturbed intents that meet the similarity threshold, i.e., when the cosine similarity between the encoded code description before and after the perturbation is greater than 0.80.



Performance of models against perturbations

Perturbation	Seq2Seq SYN	Seq2Seq SEM	Seq2Seq ROB	CodeBERT SYN	CodeBERT SEM	CodeBERT ROB	CodeT5+ SYN	CodeT5+ SEM	CodeT5+ ROB
None	0.95	0.65	-	0.93	0.69	-	0.90	0.69	-
Word Substitution	0.86	0.51	0.66	0.89	0.49	0.68	0.73	0.42	0.58
Word Omission	0.81	0.33	0.45	0.67	0.32	0.44	0.75	0.37	0.51

Syntactic Accuracy (SYN)

Indicates whether the generated code snippet is correct according to the (grammar) rules of the target language.

Semantic Accuracy (SEM)

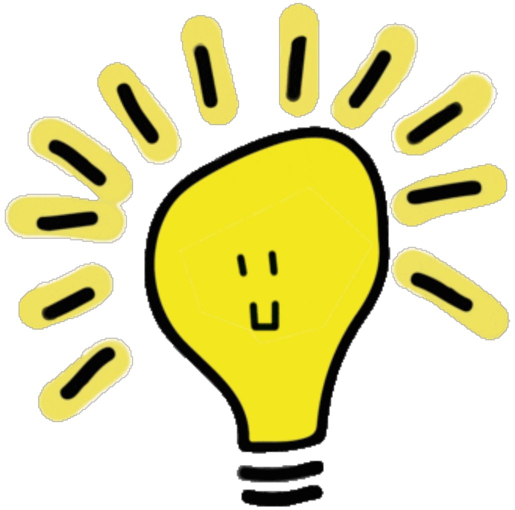
Indicates whether the output is the exact translation of the NL intent into the target programming language.

Robust Accuracy (ROB)

Evaluates the semantic correctness of the code predicted by the models before and after the perturbation.

What can we do to improve Robustness?

- **Data augmentation (DA)** refers to those techniques that **synthetically generates new training examples by perturbing existing ones in the input space**, hence increasing diversity without the need for collecting new data.
- We used DA to perturb a subset of the data used to train the models and assess if and how this technique can improve the performance of AI code generators against **new, perturbed code descriptions**.



DA Against Perturbed Code Descriptions

Perturb.	Advers. Inputs	Seq2Seq SYN	Seq2Seq SEM	Seq2Seq ROB	CodeBERT SYN	CodeBERT SEM	CodeBERT ROB	CodeT5+ SYN	CodeT5+ SEM	CodeT5+ ROB
Word Substitution	0%	0.86	0.51	0.66	0.89	0.49	0.68	0.73	0.42	0.58
	25%	0.91	0.52	0.80	0.93	0.62	0.86	0.90	0.63	0.88
	50%	0.91	0.57	0.85	0.92	0.66	0.90	0.88	0.62	0.87
	100%	0.91	0.57	0.87	0.92	0.64	0.88	0.90	0.67	0.92
Word Omission	0%	0.81	0.33	0.45	0.67	0.32	0.44	0.75	0.37	0.51
	25%	0.89	0.38	0.57	0.89	0.45	0.61	0.89	0.46	0.62
	50%	0.90	0.39	0.61	0.91	0.46	0.63	0.89	0.47	0.64
	100%	0.92	0.40	0.60	0.94	0.48	0.66	0.90	0.47	0.63

Legend

- Worst Performance
- Best Performance

Best performance when half (50% DA) of the training set or the whole training set (100% DA) is perturbed.

DIETI UNINA

MIUR DIPARTIMENTO DI ECCELLENZA 2018 2022

DIETI DIPARTIMENTO DI ECCELLENZA 2023 - 2027

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II