UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II

DOTTORATO DI RICERCA / PHD PROGRAM IN
INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Course announcement

Title: **AI Code Generation: Foundations, Evaluation, and Security**

Lecturer: **Dr. Pietro Liguori**

University of Naples Federico II
Email: pietro.liguori@unina.it

Credits: **3**

Overview

The course provides a comprehensive introduction to AI-driven code generation, covering theoretical foundations, evaluation and benchmarking methodologies, and practical case studies. Topics include offensive code generation, prompt engineering and robustness testing through data augmentation, and critical security aspects such as vulnerabilities in AI-generated code and data poisoning attacks.

There will be a final assessment.

Schedule

Lecture	Date	Time	Topics	Lecturer
1	October 7, 10:30 – 13:00	2.5 hours	Foundations of AI Code Generation	Pietro Liguori
2	October 10, 10:30 – 12:30	2 hours	Datasets and Evaluation	Pietro Liguori
3	October 14, 14:30 – 17:00	2.5 hours	AI Code Generators in Practice: Case Studies	Pietro Liguori
4	October 15, 14:30 – 16:30	2 hours	Robustness Testing & Prompt Engineering	Pietro Liguori
5	October 17, 10:30 – 12:30	2 hours	The Dark Side of AI-Generated Code	Pietro Liguori
	October 31, 14:00 – 18:00		Assessment Test	

Content details

Lesson 1 – Foundations of AI Code Generation. This lecture introduces the evolution of AI-based code generation, from early Seq2Seq models to modern large language models (LLMs). Students will learn the key concepts of attention, tokenization, and embeddings; compare encoder–decoder and decoder-only architectures; and explore training paradigms such as pre-training, fine-tuning, supervised fine-tuning (SFT), reinforcement learning with human feedback (RLHF), and parameter-efficient fine-tuning. The session also examines the differences between closed and open models, with attention to licensing and fine-tuning, and provides an overview of general-purpose and code-centric LLMs, with a brief look at Hugging Face for accessing models.

Lesson 2 – Datasets and Evaluation. This lecture focuses on how to construct and evaluate datasets for AI code generation. It covers data cleaning, duplication detection, and widely used state-of-the-art datasets. Evaluation methods range from automatic metrics to manual code review. The lecture will also explore advanced evaluation techniques, including static analysis, execution-based testing, and symbolic execution.

Lesson 3 – AI Code Generators in Practice: Case Studies. This lecture presents two domain-specific case studies: Offensive Code Generation and VHDL Code Generation from natural language descriptions. Offensive Code Generation introduces an ethical and defensive perspective on AI-driven exploit generation. Students will review responsible disclosure principles, explore categories of potential exploits at a conceptual level, learn about offensive code generation corpora, and assess performance through a guided exercise using closed-source LLMs. VHDL Code Generation illustrates AI applications in hardware description languages, covering dataset construction, model selection, and correctness verification, with a live demonstration of VHDL code generation.

Lesson 4 – Robustness Testing & Prompt Engineering. This lecture explores practical strategies to improve and stress-test AI code generation. Students will practice prompt design techniques such as few-shot, chain-of-thought, and role/persona prompting, and learn how to apply data augmentation in NL descriptions (paraphrasing, substitutions, omissions) to test model robustness. A hands-on session will guide students in prompt engineering with closed-source LLMs and in assessing the results using automatic metrics.

Lesson 5 – The Dark Side of AI-Generated Code. This lecture examines the security challenges of AI-generated software. Topics include a comparison of human-written and AI-generated code, typical vulnerability patterns and hallucinations, data-poisoning attacks, and the limitations of static analyzers and LLMs as evaluators. The session concludes with detection and remediation strategies to support secure-by-construction development workflows.

Lesson 6 – Assessment Test. Each participant will give a presentation on possible applications of AI code generators in their own research area, illustrating potential benefits and risks of their application. The presentation will stimulate critical discussion of opportunities, limitations, and possible mitigation strategies.

By October 6, 2025, participants are requested to join the Microsoft Teams group - **Team Code 3k2j9kx**

Once accepted in the Teams group, students must fill the following .xlsx file with their information (i.e., Student name and surname, e-mail, name of the PhD course, PhD cycle):

https://communitystudentiunina.sharepoint.com/:x:/s/PhDITEECourse-AICodeGeneration/EQ3XqeBo9l1KnCjQzfknby0BFIZg_Fmf-d5ckqXRRmG1og?e=eyFAPj

The course is conducted on-site in the Softel Meeting Room (Building 3, Floor 1, DIETI).

For information: Dr. Pietro Liguori (DIETI, UniNA) – pietro.liguori@unina.it *(organizer)*