

Aaditya Naik

✉ asnaik@seas.upenn.edu | [in](#) aaditya-naik | [G](#) aadityanaik | [G](#) seas.upenn.edu/~asnaik

EDUCATION

University of Pennsylvania

Ph. D., Computer and Information Science

Sept. 2020 – Present

NMIMS Mukesh Patel School of Tech. Mgmt. and Engg. (MPSTME)

B. Tech., Computer Engineering

July 2016 – May 2020

PUBLICATIONS

* Co-first author

TorchQL: A Programming Framework for Integrity Constraints in Machine Learning [📄](#)

Aaditya Naik, Adam Stein, Yinjun Wu, Mayur Naik, Eric Wong

Conditionally accepted in OOPSLA '24

Relational Query Synthesis \boxtimes **Decision Tree Learning** [📄](#)

Aaditya Naik, Aalok Thakkar, Adam Stein, Mayur Naik, Rajeev Alur

Proceedings of VLDB '24

Do Machine Learning Models Learn Statistical Rules Inferred from Data? [📄](#)

Aaditya Naik, Yinjun Wu, Mayur Naik, Eric Wong

Proceedings of ICML '23

Interactive Code Generation via Test-Driven User-Intent Formalization. [📄](#)

Shuvendu K. Lahiri*, Aaditya Naik*, Georgios Sakkas*, Piali Choudhury, Curtis von Veh, Madanlal Musuvathi, Jeevana Priya Inala, Chenglong Wang, Jianfeng Gao

Learning to Walk over Relational Graphs of Source Code. [📄](#)

Pardis Pashakhanloo, Aaditya Naik, Hanjun Dai, Petros Maniatis, Mayur Naik

Proceedings of DL4C Workshop @ ICLR '22

CodeTrek: Flexible Modeling of Code using an Extensible Relational Representation. [📄](#)

Pardis Pashakhanloo, Aaditya Naik, Yuepeng Wang, Hanjun Dai, Petros Maniatis, Mayur Naik

Proceedings of ICLR '22

Sporq: An Interactive Environment for Exploring Code Using Query-by-Example. [📄](#)

Aaditya Naik, Jonathan Mendelson, Nathaniel Sands, Yuepeng Wang, Mayur Naik, Mukund Ragothaman

Proceedings of UIST '21

Example-Guided Synthesis of Relational Queries. [📄](#)

Aalok Thakkar, Aaditya Naik, Nate Sands, Mukund Ragothaman, Mayur Naik, Rajeev Alur

Proceedings of PLDI '21

GenSynth: Synthesizing Datalog Programs without Language Bias. [📄](#)

Jonathan Mendelson*, Aaditya Naik*, Mukund Ragothaman, Mayur Naik

Proceedings of AAAI '21

Code2Inv: A Deep Learning Framework for Program Verification. [📄](#)

Xujie Si*, Aaditya Naik*, Hanjun Dai, Mayur Naik, Le Song

Proceedings of CAV '20

WORK EXPERIENCE

Microsoft Research

Summer Research Intern

June 2021 – September 2021

- Formalized the problem for interactive test-driven code generation, potential solutions and workflows, and evaluated it at scale.
- Conducted comprehensive studies of its impact on the Codex model.

University of Pennsylvania

Research Intern

Jan. 2019 – May 2020

- Worked on a project *Code2Inv* to make it compatible with various input representations including C programs and CHC constraints.
- Conducted a comprehensive study on the state-of-the-art software checkers.
- Implemented an SSA transformation for *Code2Inv* benchmarks using the *Clang C++ API*.

GetParking

Summer Intern

May 2018 – Jul. 2018

- Used transfer learning to build a deep learning model based on the InceptionV3 architecture to identify the make and model of a car given its image.
- Thoroughly reviewed existing state-of-the-art image classification models.

TEACHING EXPERIENCE

University of Pennsylvania

Teaching Assistant

May 2020 – Present

- TA for *CIS 547: Software Analysis* for Summer and Fall 2020 which covers concepts including static and dynamic analyses, symbolic executors and automated debugging.

ACM Student Chapter, MPSTME

Instructor

Sep. 2019

- Taught core C concepts to college freshman students over a 4 day workshop.

PROJECTS

Sporq

An interactive extension to VS Code for exploring code using query-by-example. It provides a flexible, easy-to-use and familiar interface to allow developers to conveniently synthesize custom program analyzers over their code.

GenSynth

gensynth.cis.upenn.edu

A genetic algorithm which synthesizes Datalog queries given a set of input and output data without requiring language biases.

Code2Inv

code2inv.org

A general end-to-end deep reinforcement learning framework which learns a valid loop invariant for any given verification task in a manner similar to how a human expert would learn the invariant.

SKILLS

Programming Languages : Python, C/C++, Bash, Java

Tools : Git, L^AT_EX, Docker

Miscellaneous : LLVM/Clang APIs, PyTorch, Keras, Z3

REFERENCES

Mayur Naik (PhD Advisor)
Professor and Graduate Chair
Computer and Information Science
University of Pennsylvania
✉ mhnaik@seas.upenn.edu
☎ 215-573-1856

Mukund Ragothaman
Assistant Professor
Department of Computer Science
University of Southern California
✉ raghotha@usc.edu
☎ 213-821-0853