Nengneng Yu

+1 484-995-8987 | ynn1999@umd.edu | College Park, MD

Education

- **University of Maryland College Park** — College Park, MD
  Doctor of Philosophy in Computer Science — *Aug 2023 - Present*
  Advisor: Zaoxing(Alan) Liu

- **Boston University** — Boston, MA
  Bachelor of Science in Computer Engineering — *Sep 2019 - May 2023*
  Magna cum Laude, Highest honors in the major

PUBLICATION & WORKS

[1] Yajie Zhou*, **Nengneng Yu*** , Zaoxing Liu, "Towards Interactive Simulacra of Internet Investigation by Human Researchers", Hot Topics in Networks (HotNets), 2023
[2] Yajie Zhou, **Nengneng Yu**, Simiao Zuo, Yue Yu, Haoming Yi, Chao Zhang, Tuo Zhao, Zaoxing Liu, "Fine-Grained, Adaptive Advanced Persistent Threats Detection with SENTINEL", Under Submission

Research Projects

- **AI-based Generative Proteomics System for Cancer Detection** — Feb 2024 – Present
  Mentor: Zaoxing(Alan) Liu, Yuefan Wang — *Froot Lab, UMD & Johns Hopkins Medicine*
  - Developed an AI system using a generative pipeline with diffusion models to generate synthetic samples, combined with XGBoost for biomarker identification and cancer stages classification.
  - Optimized data processing with distributed computing, and incorporated transfer learning and federated learning to enhance model performance and scalability.
  - Achieved near 100% accuracy in NAT/Tumor classification and improved CPTAC dataset performance by 10-20%, boosting Korean dataset from 70% to over 90% on weighted accuracy among cancer stages.
- **Interactive Research Agents for Internet Incident Investigation** — May 2023 – Present
  Mentor: Zaoxing(Alan) Liu — *Froot Lab, University of Maryland College Park*
  - Developed an LLM-based agent to simulate experienced researchers and automate the investigation process, addressing the inefficiencies of traditional manual and time-consuming Internet incident investigations.
  - Built an agent using Auto-GPT and GPT-4, equipped with autonomous information retrieval, knowledge memory, and self-learning capabilities. Tested it on challenging scenarios such as the impact of hypothetical solar storms on networks.
  - Achieved 87.5% consistency in insights compared to human experts, effectively automating complex Internet incident analysis.
- **Advanced Persistent Threat (APT) Detection and Analysis** — Feb 2022 – Present
  Mentor: Zaoxing(Alan) Liu — *Red Hat & Boston University*
  - Improve APT detection accuracy and efficiency via Tranformer-based Model.
  - Built an automated tokenizer to address system path explosion, and incorporated DAG structures with tokenized attack paths that allows to build fine-grained attack analysis instead of the simple binary classifier.
  - Outperformed existing learning based method and addressed the challenge of adapting to new attack types and helps reconstructing attack story for security analysts and engineers.

Technical Skills

Programming languages:: C++, C, Python, C#, Java
Web Technologies:: HTML, CSS, Flask, React, JavaScript
ML/AI: TensorFlow, Numpy, Pandas, Matplotlib
Miscellaneous: Linux, GDB, Git, Shell, MySQL, Latex

Services

- **University of Maryland College Park Department of Computer Science** — Aug 2023 - Present
  Graduate Teaching Assistant — *CMSC414 Network and Security, CMSC250 Discrete Structure*
- **Boston University College of Engineering** — Sep 2022 - Dec 2022
  Teaching Assistant — *EC440 Operating System*
- **Boston University College of Engineering** — Jan 2022 - May 2022
  Teaching Assistant — *EC414 Machine Learning*

Course Projects

- **Accelerate Raychasing Using OpenMP** — Sep 2023 – Dec 2023
  - The project focuses on improving the performance of the Octomap system, a prevalent mapping tool used in UAVs (Unmanned Aerial Vehicles) and robotics. The system was facing bottlenecks in its ray-tracing component.
  - The objective was to identify and alleviate the bottleneck in the Octomap system to enhance the overall performance of UAV mapping.
  - Implemented parallel computing techniques using the OpenMP library to accelerate the ray-tracing component of Octomap. This involved redesigning the original open-sourced system's workflow and data structure to allow for efficient parallel processing.
  - The acceleration of the ray-tracing component led to a significant performance improvement: up to a 60.173% increase in ray-tracing speed with a guaranteed correctness.
- **eBPF Modularity Project** — Sep 2022 – Dec 2022
  - Work with Professor from Brown University and IBM engineers to further their research on building a widely used eBPF module library.
  - Run OPENED, the tool they developed that can be used to analyze and decompose eBPF programs, on top open-source projects to extract modules.
  - Create a framework to transform the extracted module into a format with which Bumblebee tools can be run on it to create OCI images.
  - Create a framework that adds "glue logic" to allow for integration of module with L3AF/Polycube.
- **Basic Unix-like Operating System** — Jan 2022 – May 2022
  - Implemented a shell able to execute commands via system call, redirect the standard input/output (stdin/stdout) of commands to files, managing multiple commands piping, and background execution with "&".
  - User-mode-only implementation of a subset of the POSIX threads API. Allowed the creation and execution of multiple threads, as well as their scheduling in a round-robin fashion.
  - Worked on memory management at the thread level. Completed a copy-on-write (COW) thread local storage (TLS) allows threads to share data while ensuring that changes to internal data have no impact on other threads.