Siddharth Ganapathy

(424) 832-6218 | siddharthg26@berkeley.edu | linkedin.com/in/siddharthg26 | github.com/siddharthg22

Education

University of California, Berkeley May 2027 (Expected)

B.A. in Computer Science & B.A. in Linguistics

- **Computer Science Coursework:** *CS 189 (Machine Learning), CS 170 (Algorithms), CS 61B (Data Structures), CS 70 (Discrete Math), CS 61C (Computer Architecture), CS 61A (Program Structures)
- **Linguistics Coursework:** Ling 120 (Syntax), *Ling 115 (Morphology), Ling 111 (Phonology), Ling C142 (Language and Thought), Ling 100 (Intro to Linguistics)

Skills

Languages: Python, Java, C/C++, JavaScript, Swift, HTML/CSS, SQL
Libraries: NumPy, Pandas, scikit-learn, PyTorch, TensorFlow, Matplotlib, spaCy, Qiskit, Q#, lambeq
Developers Tools/Frameworks: Node.js, Express.js, MongoDB, Git, VSCode, XCode, React, React Native, JUnit

Experience

Lawrence Berkeley National Laboratory May 2024 – Present

Quantum Computing Research Affiliate *Berkeley, CA*

- Contributing to BQSKit, an open-source compiler framework for quantum computers
- Enhancing an ML-based seeded synthesis algorithm (QSeed) using PyTorch to optimize initial configurations of unitary matrices representing circuits
- Integrating QSeed with Permutation-Aware Synthesis using quantum routing algorithms and IBM Qiskit backends for validation to improve circuit depth and gate fidelities

UC Berkeley EECS Sep. 2023 – Present

Computational Game Theory Researcher *Berkeley, CA*

- Analyzing two-person abstract strategy board games through the development of deterministic solvers
- Developed a solver for Dino Dodgem that un-hashed ∼39K possible moves into Win/Lose/Tie moves, as well as a Graphical User Interface (GUI) using Python
- Currently exploring database compression to support solvers for computationally-large games and helping new members develop rudimentary solvers in C

Quantum Computing @ Berkeley Sep. 2023 – Present

Quantum Natural Language Processing (QNLP) Researcher *Berkeley, CA*

- Helped develop a Quantum Recurrent Neural Network (QRNN) for conducting common sentiment analysis tasks
- Utilized lambeq to convert input sentences into quantum circuits to be parameterized and fine-tuned using SPSA
- Implemented QRNN cells using amplitude amplification and RUS architecture to reduce Clifford gate count, optimizing circuit performance by 10.1%
- Achieved training accuracy of 69.6% on single-parameterized cells on lambeq embedding compared to 63.9% on classical embedding

Projects

Intelligent Document Insights | *Palantir AIP, Python, React, GPT-4 API, Ontology SDK* Nov. 2024 – Present

- Developing a data pipeline using Palantir's AI Platform and Ontology SDK to ingest, process, and classify unstructured documents
- Integrating GPT-4 API to perform summarization, action item extraction, and Named Entity Recognition (NER) for generating concise insights, with a focus on Semantic Role Labeling
- Engineering a front end dashboard using React to display document summaries via interactive visualizations

Carpe Scientiam | *Swift, MongoDB, JS, Node.js, Express.js, Figma* Aug. 2023 – Present

- Developing a full-stack iOS mobile app for Latin learners, featuring exam prep, personalized quizzes, and word parsing with Latin WordNet API
- Created storyboards with Figma, designed front-end with Swift, and implemented REST API endpoints for server-side functionality with Express.js and Node.js
- Engineered back-end with MongoDB to enable storage, retrieval, and management of quiz results and study material
- Conducted beta testing using TestFlight to understand user experience and improve app performance

Build Your Own World (BYOW) | *Java, JUnit, IntelliJ, TileEngine*

- Developed a 2D tile-based world exploration engine using Java
- Implemented random world generation with RandomUtils, world saving and loading with Serialization, and graphics rendering with StdDraw
- Introduced features such as mob generation/item dispersal with RandomUtils, ability to shoot, and in-game time
- Integrated unit tests using JUnit to evaluate core game mechanics, edge cases in world generation, and stability