Neo Lee

neo.lky852@gmail.com | (213) 713-0311 | github.com/moneybabe

Education

University of California, Berkeley **GPA: 4.0/4.0**
Bachelor of Arts, Applied Mathematics, Computer Science *Graduation: Spring 2025*
Cal Alumni Leadership Scholarship

Relevant Coursework: Machine Learning, Optimization Theory, Time Series Analysis, Stochastic Processes, Probability Theory, Linear Algebra, Discrete Mathematics, Graph Theory, Real Analysis, Numerical Analysis, Cryptography, Data Structures and Algorithms, Functional Programming, Object Oriented Programming, Dynamic Programming

Technical Skills

Languages: Cantonese, Mandarin, English, Python, Java, SQL, MATLAB, R, Javascript, HTML, CSS, LaTeX
Tools: Pytorch, Sklearn, *Boost*, Statsmodels, Pandas, Numpy, Plotly, Matplotlib, MySQL, Selenium, BeautifulSoup, web3.py, Gcloud SDK, Docker, Git, Bash

Experience

CyphaLab **Remote**
Data Science Intern *Januaray 2024 - Present*

- Ensembled CodeBERT and gradient boosting decision trees using Pytorch and XGBoost to classify malicious Ethereum smart contracts, achieving 0.89 F1 score on the test set.
- Built a transaction optimizer with Python to dynamically calculate the optimal gas price for Ethereum transactions and routing them through the optimal MEV relayer, reducing gas fees by 30%.
- Built a data relayer hosted on Google Cloud Virtual Machine to provide on-chain DEX transaction data for online time series forecasting.

UC Berkeley Department of Mathematics **Berkeley, CA**
Undergraduate Researcher - Stake-governed Random Turn Games *August 2023 - Present*

- Built a finite integer line tug-of-war game simulator with Python, Numpy, and Pandas to solve for Markov perfect equilibria with dynamic programming, and visualized the results with Matplotlib.
- Constructed a computer assisted proof for the sufficient and necessary condition for the existence of a Markov perfect equilibrium in infinite integer line tug-of-war games, being that the reward ratio is bounded within a 1×10^{-4} interval from a symmetric game: paper is currently under review.
- Reduced the run-time of the computer-assisted proof by 60% through the implementation of dynamic programming and vectorization techniques.

Undergraduate Researcher - Mechanistic Interpretability *September 2023 - Present*

- Reverse engineered Stockfish's efficiently updatable neural network's learned algorithm using Pytorch, Sklearn, and Seaborn, achieving MSE of 1.8 compared to a simple linear regression model with MSE of 126.
- Applied dimension reduction techniques such as SVD, neuron pruning, and feature projection onto ReLU privileged basis to reverse engineer the embedding layer, showing that each neuron's activation space is 95%+ correlated.
- Aggregated 38GB training dataset with Sqlite and used Git for version control.
- Built an Alpha-beta pruning algorithm & NN based chess engine with Python to study the effect of neural network based evaluation functions on the performance of the algorithm.

UC Berkeley Department of EECS **Berkeley, CA**
Academic Tutor - CS61A *August 2023 - Present*

- Tutored students in Functional Programming, Object Oriented Programming, and Dynamic Programming with Python in lab sessions.
- Held weekly office hours to help students with homework and projects, and tutored other tutors.

Projects

Trading Bot *June 2023 - August 2023*

- Implemented machine learning models with Pytorch, Statsmodels, and XGBoost for crypto price forecasting, e.g. LSTM, ARIMA, ETS Smoothing, Multi-linear Regression, GBDT, BERT.
- Built a trading bot with Python to trade cryptocurrency on Bybit using the Bybit API, achieving 2.98 Sharpe Ratio on the test set by performing time series cross validation and features engineering.