Usman Khan

usmankhan.dev | usman@usmankhan.dev | linkedin.com/in/khanu | github.com/ukhan1219

Education

University of Central Florida — Orlando, Florida

B.S. in Computer Science 3.8/4.0 GPA — *Expected Graduation: December 2025*

Relevant Coursework: *Algorithms in Machine Learning, Artificial Intelligence/Machine Learning, Robot Vision, Matrix and Linear Algebra* Calculus, Statistics, Analysis of Computer Networks, Topics in Cybersecurity

Technical Skills

Languages: Python, Java, OCaml, C++, TypeScript, C, JavaScript, SQL, PostgreSQL, Neo4J, R, PHP, HTML, CSS
Frameworks: PyTorch, Keras, TensorFlow, NumPy, Pandas, MatPlotLib, SKLearn, Next.js, React, Node.js, tRPC, Tailwind
Tools: Git, Github, Docker, Vercel, Linux, LaTeX, Prisma, Heroku, Figma, Amazon Web Services, Google Cloud Platform

Work Experience

Machine Learning/AI Researcher — April 2024 - Present

University of Central Florida — Python, TensorFlow, PyTorch, NumPy, SKLearn, NetworkX, Pandas — *Orlando, Florida*

- Applied **AI, machine learning,** and **statistical inference** to automate large-scale entity metadata analysis, focusing on detecting foreign ownership and influence through **Neo4J** database queries.
- Developed **temporal prediction models** for company longevity using **RandomForestRegressor** and **NetworkX** to classify edge and node relationships, advancing corporate structure research.
- Designed algorithms to process extensive datasets, **increasing accuracy and speed** of entity tracking by **30%**

Software Engineering Intern — January 2024 - August 2024

DrivnBye — TypeScript, Next.JS, React.JS, HTML, CSS, Prisma — *Orlando, Florida*

- Spearheaded backend development and **resolved critical bugs** in DrivnBye's content moderation platform, ensuring a **seamless user experience** for moderators.
- Implemented features like reporting tools for accounts, posts, and comments, as well as banning functionalities, **improving platform compliance** by **25%.**
- Contributed to creating a safer and more engaging environment, **enhancing content moderation** capabilities for **5,000+ active users.**

Projects

StockBot | *Python, PineScript, TensorFlow, Keras, PyTorch, Pandas, Numpy, SKLearn, CRON*

- Engineered a **Markov Chain Monte Carlo** model with **Metropolis-Hastings** sampling to provide real-time trading strategies for S&P 500 e-mini futures, integrated with **TradingViews** strategy tester.
- Optimized data pipelines to fetch, clean, and normalize stock data using **Yahoo Finance API** and **SKLearn**, **improving model accuracy by 20%.**
- Trained a neural network using **TensorFlow** and **Keras** with early stopping and dropout layers, achieving a **15% improvement** in trading signal **precision.**

Glance | *t3 Stack (Next.js, React, tRPC, TypeScript, Prisma, Tailwind CSS, PostgreSQL), Gemini AI, Plaid, Polygon, Heroku*

- Spearheaded development for a **financial management** application using the t3 stack. Designed in **Figma**, wrote database schema, and hosted on **Heroku**. Integrated **Plaid**, **Gemini**, and **Polygon** APIs, delivering **investment insights**, budgeting tips, and a financial overview.
- Supplied **secure authentication** via **NextAuth** and integrated financial accounts with Plaid API to retrieve transactions and balance data. **Engineered prompts** for Gemini AI to provide **personalized financial insights**, resulting in a **25% improvement** in investment recommendations and spending optimization.
- Constructed **tRPC endpoints** for seamless frontend-backend communication, boosting development **efficiency by 40%**. Increased API **response times by 35%**. Engineered database interactions to persist user data and **financial analysis**, **enhancing user experience** across sessions.

CETA | *Python, Pandas, SKLearn, Numpy, TensorFlow, PyTorch, NetworkX, PySage, GraphSage*

- Designed and implemented predictive models for temporal corporate analysis, employing **RandomForestRegressor** and **advanced NetworkX** graph analytics to **enhance predictive accuracy** for research needs.
- Enhanced data preprocessing techniques (train-test split, label encoding), **ensuring model validity and accuracy.**
- Utilized **NetworkX** to analyze **complex graph structures**, extracting relevant attributes for analysis from a **GraphML** file.