**Pranav Goyanka**

pgoyanka@gmail.com | (774) 284-6311 | Boston, MA | [github.com/pranavgoyanka](https://github.com/pranavgoyanka) |[linkedin.com/in/pranavgoyanka/](https://linkedin.com/in/pranavgoyanka)

**EDUCATION**

**Boston University Dec 2024**

MS in Computer Science | GPA: 3.78/4.00 Boston, MA

Courses: Principles of Machine Learning, Distributed Systems, Tools for Data Science, Graduate Computer Networks

**Thapar University Jun 2021**

BE in Electronics and Communication Engineering | CGPA: 3.73/4.00 Patiala, India

Courses: Data Structures and Algorithms, Operating Systems

**SKILLS**

* **Programming Languages:** C++, C, Python, Java, Go, TypeScript, JavaScript, SQL, HTML/CSS
* **Frameworks:** PyTorch, TensorFlow, Docker, Node.js, Socket.IO, WebSocket, OpenTelemetry, gRPC, Flask
* **Tools and Libraries:** Docker, Apache Flink, Kafka, Redis, scikit-learn, AWS, RESTful API, Git, Linux, DynamoDB
* **Other Skills:** \_\_\_\_

**EXPERIENCE**

**Graduate Teaching Assistant and Course Designer Jan 2024 – Present**

Boston University Boston, MA

* Implemented the OmniPaxos consensus protocol and developed over **30 unit-tests in Go**.
* Designed assignments, grading infrastructure, and coursework for writing formal specifications using TLA+.
* Conducted weekly lab sessions and office hours for the courses CS350 and CS651 taught by Prof. John Liagouris.

**Software Development Engineer Oct 2022 – Jul 2023**

Mobile Premier League Bangalore, India

* Achieved a **40% reduction in infrastructure costs** and utilization by implementing a library for metrics collection and auto-scaling using OpenTelemetry, enabling graceful node shutdowns and adoption multiple cross-functional teams.
* **Boosted user engagement and retention by 70%** by expanding matchmaking systems with cross-country support, enabling seamless interactions across international user bases.
* Enabled faster development and reduced bugs by **engineering backend systems and libraries** with extensive end-to-end testing for **Node.js microservice** based server-authoritative games, eliminating boilerplate code across 7 games.

**Software Development Engineer Jan 2021 – Oct 2022**

Amadeus Software Labs Bangalore, India

* **Reduced chatbot** **development** **effort** **by over 50%,** by accelerating bootstrapping time, by creating ‘Chatbot as a Service’, a modular Java framework using Spring Boot for NLP APIs and database APIsused by over 5 teams.
* **Reduced incidents by 40%** by enhancing the stability, recovery mechanisms and **regression tests** of the **C++ based** **backend** – the Back Office tool, to comply with the IATA NDC standards.

**Software Developer Jun 2020 – Aug 2020**

Google Summer of Code Remote

* Selected for **GSoC** as a part of the 18% applicants globally and **contributed to the open-source** project ‘Social Street Smart’, aimed at combatting misinformation and fake news.
* Generated and **deployed** serverless **Machine Learning** **models**, **CI/CD pipelines**, **and APIs** for fake news detection.
* Reduced model size by 85% by migrating TensorFlow machine learning models to TFLite; hosted them on AWS Lambda.

**PROJECTS**

**Retrieval-Augmented Generation for Internal Documentation Jul 2024 – Aug 2024**

* Developed a **RAG pipeline** that optimizes LLM responses based on proprietary documentation.
* Created a **user-friendly web UI using Flask** for uploading documentation and interacting with the model.
* Evaluated the correctness and accuracy of responses across 5 different LLMs with RAG enabled and disabled.

**Automated Trading System Mar 2024 – Apr 2024**

* Predicted daily temperatures using **LSTM models** and performed automated trading with **over 80% accuracy.**
* Collected, cleaned, and processed weather data with over **12,000 data points from 4 sources** via APIs for model training.

**Flink on the Edge** **Jan 2024 – May 2024**

* Added heterogeneous device support to Apache Flink for enabling **Edge compute on geo-distributed queries**.
* Built a system for dynamically offloading intensive tasks to edge nodes **to minimize overall latency.**
* Developed **a Docker environment to simulate network conditions** for running experiments and benchmarking.

**Fault Tolerant Key-Value Store Oct 2023 – Nov 2023**

* Built a **scalable key-value storage service** by implementing the **Raft** distributed consensus algorithm in Go.
* Ensured robustness against network and node failures by using a **comprehensive suite of over 40 unit-tests.**