Parham Moradi

(647)513-3618 p3moradi@uwaterloo.ca LinkedIn Github Portfolio

Skills

Programming & Frameworks: Python, SQL, TensorFlow, scikit-learn, PyTorch, Pandas, Seaborn
Data & AI Techniques: Agentic AI, RAG , NLP, RL, Optimization, Data Modeling, Predictive AI
Tools & Technologies: Looker, Tableau, Power BI, Langchain, Flask, Heroku, Docker, Kubernetes, SQLite
Operating Systems and subsystems: Ubuntu, WSL, Windows

A strategic problem-solver who excels at translating mathematical concepts into practical solutions.

EDUCATION

MASc Management and Data Sciences Waterloo, Canada
University of Waterloo Sept.2021 - Feb.2024

BSc Computer Science and B.Sc. Electrical Engineering Tehran, Iran
Sharif University of Tech Sept.2016 - Sept.2021

Experiences

AI/ML Product Analyst

Doctly.ai (2025–Present)

- Designed a **PDF/photo-to-Markdown benchmarking framework** to evaluate **Doctly.ai's generative AI** extraction performance.
- Engineered **testmark**, a robust performance evaluation tool measuring **throughput**, **latency**, **token usage**, and **accuracy** across 10 LLMs on **1,000+ documents**.
- Developed backend infrastructure with **PostgreSQL** and **Azure Storage** to manage datasets, host documents, and track model predictions and evaluations.
- Authored an **article** on benchmarking analysis and methodology evaluation, scheduled for publication on **Medium** on May 3.
- Conducted comparative analysis of LLMs—**GPT-4o, Gemini 2.5-flash, Google Document AI, Azure Form Recognizer, AWS Textract, Mistral AI**, and **Doctly**—to benchmark extraction accuracy and output consistency in Markdown and JSON.
- Built generative AI pipelines using **Python, JavaScript**, and **TypeScript**, and integrated with **Hugging Face** and **PostgreSQL** for data and model management.
- Achieved **91%+ Markdown extraction accuracy** with doctly agent, outperforming GPT-4o and Gemini on benchmark datasets.

Data Scientist

University of Waterloo (2021-2024)
Research Assistant **Project Showcase**

- Collaborated with **15 Canadian hospitals** to **optimize healthcare data workflows**.
- Worked as part of three cross-functional teams to deliver results at weekly meetings.
- Defined the goal of improving patient scheduling performance and discussed the hospital objectives, which led to **data-driven modeling**.
- On 15 Historical Data Sets (with **50 features** and **25,000 records**) applied **preprocessing** methods for the data analysis preparation.
- Conducted **quantitative analysis** using **causal inferences**, **statistical analysis**, **hypothesis testing**, **A/B testing**, **simulation analysis**, and **sensitivity analysis** to drive optimized data-driven decisions.
- Developed and implemented **AI/ML models**, including **predictive modeling**, **generative AI**, **optimization algorithms**, and **reinforcement learning** techniques for improving the decision-making performance.
- Analyzed Datasets by Python libraries such as **NumPy** and **Pandas** for data manipulation, and **PyTorch** for deep learning model development.
- By **exploratory data analysis (EDA)** uncovered **hidden patterns** in long waiting times and queue bottlenecks.
- Designed and managed **relational databases** using **MySQL** and **SQLite**.

- Using **counter factual analysis** and **Agentic AI** suggested better decisions that optimized hospital workflow processes, reducing total patient **waiting times by 30%** and improving decision-making **performance by 70%** through data-driven strategies.
- Used **Power BI**, **Matplotlib**, and **Seaborn** for visualizing findings and performance metrics.
- At each step, translated mathematical concepts into actionable insights for non-technical stockholders.

Data Analytics (Intern)
RastakMS (2020)

R&D team

- Conducted research and analysis of **demand planning** and **probability of default** for the **carchain project**, leading to actionable insights for scalability improvements.
- Used **Fraud Analytics** and identified security vulnerabilities in the Carchain project such as Data availability attacks.

Projects

Data-Driven Patient Scheduling with AI and RL
Showcase

Master's Thesis — University of Waterloo

- Built the **optimized quantitative** patient scheduling model.
- Checked the robustness of the algorithm using **sensitivity analysis**.
- Analyzed Canadian hospitals' datasets and estimated decision-making policies using **Python**, with data processed in JSON and CSV formats.
- Built a **predictive AI-based** patient scheduler to **test hypotheses** using **A/B testing method**, improving the total waiting times by 30%.
- Used **Mathematical Optimization, Reinforcement Learning (RL)** and **Pandas** to improve decision-making performance by 70%.

YouTube API Chatbot – Answers Questions from Channel Content
Demo on my Website **GitHub Repository**

- Built an intelligent chatbot to extract and process video content from any public channel, using **information retrieval** and **deep seek API**.
- **Extracted** video titles, descriptions, and captions from a YouTube channel URL and saved in a **SQLite** database.
- Chatbot answers **user queries** about the channel, including speakers, key topics discussed, and specific mentions across videos.
- Applied **prompt engineering** and **Retrieval-Augmented Generation (RAG)** to answer user queries accurately and prevent LLM hallucinations.
- **Deployed a personalized version** of the chatbot on my **Portfolio Website** to answer questions about my projects and experience.
- Developed the backend using **Flask** and deployed on **Heroku** for scalable web hosting.
- Integrated **LangChain** for dynamic retrieval-augmented generation pipelines and used **Hugging Face Transformers** for **NLP tasks**.
- Optimized **semantic search** and document similarity with **FAISS**, deployed via **Hugging Face Transfer**.
- Use case includes educational platforms, news summarization, and efficient browsing of technical content.

Building a Scalable Machine Learning API
GitHub Repository

- Built a user-friendly **ML API** that enables non-expert users to **upload datasets, train models, save best-performing models, and receive predictions and evaluation scores.**
- Saved the best-performing model to **SQLite database**.
- Validated endpoints using **Postman and cURL.**
- **Dockerfile** ensures seamless local deployment.
- Developed **Kubernetes** deployment and service YAML files.
- Deployed the app locally using **Minikube**.
- README.md with Docker setup instructions, API usage examples, and **CI/CD pipeline guidance**..

Coursework

Machine Learning with Python
IBM Certificate

- Linear and Logistic Regression, Sigmoid Function, Gradient Descent, SVM, Snap ML, Grid Search, and Clustering.