**Bala Showri Jeevan Reddy Arlagadda**

[A blue hexagon with white text

AI-generated content may be incorrect.](https://www.credly.com/badges/0a94e000-3e04-4c28-aeee-b2d0fcdaf047/public_url)

**Python – Backend and AI/ML Developer**

[**balashowrijeevan4@gmail.com**](mailto:balashowrijeevan4@gmail.com)**| +1 (352)-363-1174 | [LinkedIn](https://www.linkedin.com/in/arlagadda-jeevan-reddy/) |** [**portfolio**](https://portfolio-project-n1rj.onrender.com/portfolio) **|** [**medium**](https://medium.com/@jeevanr.arlagadda)

**Profile**

* **AWS - certified** Python Developer with **7+** years of expertise in backend development and machine learning, focusing on building scalable, high-performance systems. Proficient in leveraging Python frameworks and tools to design and implement intelligent, data-driven solutions.
* Specialized in the complete machine learning lifecycle, including data preprocessing, feature engineering, model training, hyperparameter tuning, and deployment. Skilled in using TensorFlow, PyTorch, and Scikit-learn to develop predictive models for inventory optimization, demand forecasting, anomaly detection, and other applications.
* Proficient in managing large datasets with Python, SQL, and AWS tools, performing data integration, transformation, and querying to ensure high-quality, actionable data pipelines.
* Experienced in deploying machine learning models to production environments, integrating them seamlessly into backend systems through REST APIs, and leveraging scalable cloud infrastructure.
* Expertise in Python libraries such as BeautifulSoup, NumPy, SciPy, Matplotlib, Pandas, Seaborn, and NetworkX, with strong database skills in MySQL. Skilled in using IDEs like PyCharm, Visual Studio code, and Spyder for efficient development workflows.
* Designed and implemented robust backend architectures optimized for scalability and performance, integrating machine learning models and APIs for real-time decision-making systems.
* Built and deployed AWS Lambda functions to enable serverless workflows triggered by data changes and user actions, enhancing scalability. Experienced in cloud platforms such as AWS and GCP, optimizing deployments for efficiency and reliability.
* Skilled in creating data pipelines with tools like Apache Kafka and Redis to ensure real-time insights and fast data retrieval.
* Familiar with advanced machine learning techniques, including NLP for sentiment analysis, time-series modeling for forecasting, and ensemble methods such as Random Forests and Gradient Boosting.
* Collaborates effectively with cross-functional teams to deliver innovative solutions, leveraging both backend and machine learning expertise to drive business impact. Committed to staying updated on the latest advancements in machine learning and backend technologies.

**Technical Skills**

**Programming Languages:** Python, C/C++, JavaScript

**Database/OS:** MySQL, PostgreSQL, Snowflake, Oracle, Linux/Unix Systems

**Data Processing & ETL:** ETL Pipelines, Data Annotation, Data Labeling, Synthetic Data Generation

**Frameworks:** TensorFlow, Keras, Scipy, SpaCy, Scikit-learn/sklearn, Pandas, NumPy, Matplotlib, Seaborn

**AI Development:** Deep Learning, Machine Learning, AI Model Training, Fine-Tuning, Bias Detection, Ranking, Optimization, NLP, CNNs, Transformers, BERT, LLM-based Models, AI Dataset Management, Data Preprocessing, Feature Engineering

**Web Development:** Flask, Django, RESTful Web Services, GraphQL Development, Web Sockets, Asynchronous Processing

**Tools/Software:** Tableau, H20.ai, Chart.js, D3.js, Power BI, Pytorch, JSON, GitHub

**Development platform:** Visual studio code, PyCharm, Sublime text, Spyder, Google Colab, Jupyter notebook

**Cloud platform:** AWS (ECS, Lambda, SageMaker), Google Cloud Platform (Vertex AI), Azure

**Specializations:** Object-Oriented Programming, Deep Learning, Machine Learning

**Monitoring:** Prometheus, Grafana

**Data Skills:** Data visualization, Data analysis techniques

**Work experience**

**Client: FactSet - Norwalk, Connecticut, USA February 2023 - Present**

**Role: Sr. Software Developer**

**Project Description:**

Developed an AI-Driven Dynamic Pricing Engine for e-commerce platforms, utilizing Scikit-learn and XGBoost to predict optimal prices based on historical sales, demand patterns, and inventory levels. Integrated reinforcement learning for real-time pricing adjustments, enhancing revenue and competitiveness. Built a RESTful backend with Django REST Framework and used Beautiful Soup and Scrapy to scrape competitor pricing data, automated via scheduled jobs. Enabled real-time inventory and sales tracking with WebSocket-based updates and improved performance through Redis caching. Deployed containerized services using Docker on AWS ECS with auto-scaling and monitored performance with AWS CloudWatch. Optimized MySQL for storing product data, implementing replication, indexing, and ETL pipelines for efficient query handling. This scalable solution ensured real-time pricing updates, reduced latency, and enhanced system reliability.

**Responsibilities:**

* Utilized Scikit-learn and XGBoost to train regression models predicting optimal prices based on historical sales, demand patterns, and inventory levels.
* Integrated reinforcement learning to dynamically adjust pricing strategies based on real-time feedback from sales performance.
* Built a RESTful backend service using Django REST Framework to expose APIs for fetching and updating product prices.
* Designed modular components for managing pricing logic, integrating machine learning predictions, and handling API requests.
* Used Beautiful Soup and Scrapy to scrape competitor pricing data from e-commerce platforms.
* Scheduled scraping jobs with Python’s schedule library to collect competitor data at regular intervals.
* Implemented real-time inventory and sales tracking using WebSocket-based updates between the inventory management system and the pricing engine.
* Cached frequent API responses with Redis to reduce latency and improve system performance.
* Deployed containerized services using Docker and managed them on AWS ECS with auto-scaling enabled for handling traffic spikes.
* Configured AWS CloudWatch to monitor service performance and generate alerts for anomalies.
* Used MySQL for storing product details, sales trends, competitor pricing data, and model predictions.
* Designed tables with optimized indexing for high-speed queries on frequently accessed data, like product sales and pricing history.
* Created ETL pipelines to import competitor pricing and sales data into MySQL at scheduled intervals.
* Used AWS Data Pipeline for transforming and loading data into the database, ensuring consistency across large datasets.
* Developed stored procedures for aggregating sales and pricing data to generate summary statistics, such as average sales per product and price elasticity.
* Set up MySQL replication to support read-heavy operations, ensuring consistent availability of real-time pricing data for the frontend.
* Enabled failover mechanisms to maintain system uptime during high-traffic events.

**Client:** **Legacy Health - Atlanta, GA, USA April 2021 - January 2023**

**Role: Software Developer**

**Project Description:**

Developed a scalable AI training system to improve the accuracy, reliability, and contextual understanding of AI-generated code responses. Designed automated evaluation and ranking processes to assess AI outputs, identify inconsistencies, and refine learning models. Built a structured training workflow that continuously adapts based on real-time feedback, ensuring high-quality and bias-free AI responses. Integrated data annotation and optimization techniques to enhance model performance and maintain consistency. Enabled seamless model updates and fine-tuning, allowing AI to evolve with new data and improve over time. Deployed a monitoring system to track AI performance, ensuring continuous improvements and reducing errors. Automated model deployment and versioning workflows, maintaining a stable and efficient AI training process. This solution provided a structured, self-improving AI system that delivers more precise and relevant coding responses.

**Responsibilities:**

* Designed and developed AI model training pipelines leveraging PyTorch, TensorFlow, and Scikit-learn, optimizing model learning from structured datasets.
* Engineered an AI model evaluation framework that analyzes, ranks, and scores AI-generated code responses, improving precision and reducing hallucination errors.
* Implemented fine-tuning strategies using hyperparameter optimization (Grid Search, Bayesian Optimization, and AutoML) to improve AI performance.
* Developed RESTful and GraphQL APIs for AI model interaction, allowing seamless data annotation, evaluation, and automated retraining workflows.
* Automated dataset preprocessing and curation by implementing synthetic data generation, feature engineering, and data augmentation techniques.
* Optimized AI model inference speeds by implementing asynchronous processing, multi-threaded API handling, and Redis-based caching, reducing response latency by 45%.
* Integrated AI bias detection algorithms, enabling fairness auditing through SHAP, LIME, and adversarial testing techniques to detect inconsistencies.
* Developed and monitored AI performance dashboards using Prometheus, Grafana, and ELK Stack, tracking model drift, response quality, and dataset variations.
* Implemented real-time feedback loops for AI model adaptation, leveraging active learning and reinforcement learning techniques to enhance accuracy dynamically.
* Built scalable CI/CD pipelines for automated AI model deployment, versioning, and rollback strategies, ensuring continuous improvement and minimal downtime.
* Deployed AI models in cloud environments (AWS SageMaker, GCP Vertex AI) with auto-scaling capabilities, GPU acceleration, and containerized deployments.
* Integrated large-scale dataset annotation workflows, improving data consistency and reducing annotation errors through automated validation checks.
* Optimized model training and serving costs by implementing efficient batch processing, cloud resource allocation, and model distillation techniques.
* Developed AI model benchmarking scripts, comparing performance across different architectures (LSTMs, Transformers, CNNs) to determine the best-fit models.
* Ensured security and compliance by implementing RBAC, encryption mechanisms (AES-256, TLS 1.2+), and audit logging for AI model interactions.

**Client:** **Trek Bikes - Waterloo WI, USA February 2019 - March 2021  
Role: Software Developer**

**Project Description:**

Developed a Predictive Maintenance System for Industrial IoT to optimize equipment uptime and reduce maintenance costs. The system analyzed sensor data from machinery using TensorFlow and Scikit-learn to predict failures, schedule maintenance, and minimize downtime. Built with FastAPI and PostgreSQL, the platform provided real-time analytics, automated alerts via Redis Pub/Sub, and scalable deployment on AWS ECS with Kubernetes. Ensured secure data handling with RBAC and AES-256 encryption, while leveraging Prometheus and Grafana for performance monitoring.

**Responsibilities:**

* Designed RESTful APIs using FastAPI to ingest and process real-time sensor data (e.g., temperature, vibration) from IoT devices.
* Trained time-series forecasting models with TensorFlow and Scikit-learn to predict equipment failures, achieving 95% prediction accuracy.
* Implemented anomaly detection using PyTorch to identify irregular patterns in sensor data, triggering maintenance workflows.
* Optimized PostgreSQL with TimescaleDB for time-series data storage, improving query speeds by 35% through indexing and partitioning.
* Cached high-frequency sensor metadata using Redis to reduce API latency by 50% during peak data ingestion.
* Containerized services with Docker and orchestrated deployments on AWS ECS/Kubernetes for auto-scaling during operational spikes.
* Secured data pipelines with JWT authentication and RBAC to restrict access to authorized engineers and systems.
* Automated ETL workflows with Pandas and NumPy to cleanse and normalize raw sensor data, enhancing model reliability.
* Deployed serverless data preprocessing tasks on AWS Lambda to handle batch jobs cost-effectively.
* Built CI/CD pipelines using GitHub Actions for seamless testing and deployment of model updates.
* Monitored system health with Prometheus and visualized metrics (e.g., API latency, error rates) via Grafana dashboards.
* Conducted load testing with Locust to validate scalability under 15,000+ concurrent sensor data streams.
* Integrated Redis Pub/Sub for real-time alerts to maintenance teams when critical thresholds were breached.
* Collaborated with cross-functional teams to ensure alignment between predictive models and operational requirements.

**Client:** **ResMed, Hyderabad, India July 2017 - January 2019**

**Role: Software Developer**

**Project Description:**

Built a multi-tenant SaaS platform to support multiple clients with data separation using a smart database design. Designed APIs to allow client-specific features and faster response times with improved data caching. Used cloud services to ensure the system could grow with demand and stay reliable. Added tools to monitor performance and quickly fix any issues. Automated updates and tested the system to handle heavy traffic smoothly. Made it easy to onboard new clients and keep the system running without interruptions

**Responsibilities:**

* Used FastAPI to build lightweight, fast, and scalable APIs.
* Implemented role-based access control (RBAC) and tenant-specific configurations via modular service architecture.
* Integrated Redis for caching frequently accessed tenant-specific data to enhance response times.
* Dockerized backend services for isolated and reproducible environments.
* Orchestrated services using Kubernetes for scalability and fault tolerance.
* Designed PostgreSQL schemas to isolate tenant data securely and efficiently.
* Optimized queries with indexing for high-performance data retrieval in multi-tenant environments.
* Configured PostgreSQL replication for high availability and reduced downtime.
* Automated schema creation and tenant onboarding using ETL pipelines for seamless operations.
* Deployed containerized services on AWS ECS with auto-scaling capabilities to handle varying tenant traffic.
* Set up monitoring and alerting systems with Prometheus and Grafana for real-time metrics and issue resolution.
* Automated CI/CD pipelines using GitHub Actions for continuous testing and deployment.
* Conducted performance testing using JMeter to ensure reliability under high traffic loads.
* Enabled failover mechanisms and elastic scaling to maintain uptime and responsiveness during traffic spikes

**EDUCATION**

**University of Florida, Gainesville** **Florida, USA**

*Master of Science, Computer Science and Engineering*

**Amity University, Noida** **Uttar Pradesh, INDIA**

*Bachelor of Technology, Artificial Intelligence*