Matthew Gardner

Data Engineer/ML Engineer & Data Scientist

(205) 417-7473 **matthewgardner717@gmail.com** Bedias, TX

Summary/Objective

Highly skilled Data Engineer with over **10 years** of **hands-on data-based experience** specializing in designing, building, optimizing & managing end-to-end; Implementing advanced AI, Machine Learning and Data Analysis solution.

Adept at leveraging tools like Terraform, Docker, Kubernetes and MLOps to ensure seamless CI/CD workflows and infrastructure as code practices while incorporating industry best practices.

Demonstrated ability to lead **cross-functional teams**, **mentor junior data engineers/data scientists**, and junior **data scientists** in delivering **high-quality data-driven projects**.

Highly motivated & adaptive professionals committed to continuous learning & growth, consistently seeking to explore & embrace **new technologies** & environments to drive solutions & stay at the forefront of industry trends.

Passionate about delivering high-impact results via research, **technical expertise**, innovation, **collaboration**, & a **problem-solving acumen** with a continuous drive for staying ahead in the rapidly evolving data landscape.

Education

University of Texas

Bachelor of Computer Science, 2009 - 2013

Austin, TX

Skills

- **Language**: Python, SQL, Scala, Shell Scripting, Javascript, PHP
- **Cloud Platforms**: AWS (Redshift, Lambda, S3 etc.), Azure & Google Cloud Platform (BigQuery, DataFlow)
- **Data Engineering**: ETL/ELT, Data Pipelines, Spark, Kafka, Hive, Sqoop, Flume etc.
- **DevOps & Infrastructure**: Kubernetes, Docker, Terraform, Jenkins, GitHub Actions etc.
- **Monitoring & Logging**: CloudWatch, Airflow Metrics, Prometheus, Grafana, Datadog, ELK Stack
- **Databases**: MongoDB, PostgreSQL, Oracle, Cassandra, SQL/NoSQL, Kafka Schema Registry
- **Machine Learning & AI**: MLOps, DVC, Model Deployment, Pandas, Matplotlib, Scikit-Learn, PyTorch
- **Data Modeling**: Dimensional modeling, Star Schema, Snowflake Schema, Data Vault etc.
- **Data Warehousing**: Snowflake, BigQuery, Redshift and Synapse Analytics
- **Visualization Tools**: PowerBI, Tableau, Looker, Seaborn, Matplotlib, Excel
- **Testing**: Great Expectations, Unit Tests for Data Pipelines
- **Streaming**: Apache Flink, Apache Kafka, Amazon Kinesis
- **Familiar with**: Web Scraping, Tensorflow Extended(TFX), MLOps Frameworks, Version Control, Data Governance, Git, Apache Superset etc.

Work Experience

DATA ENGINEER LEAD

Infosys | Raleigh, NC (Remote) | **Aug 2019 – Present**

- Designed & optimized **ETL processes on AWS platforms** like Glue, EMR & S3 to handle large-scale structured & unstructured datasets.
- Developed **Spark applications** to improve the efficiency & scalability of **batch data processing** pipelines.
- Built & managed **Databricks Lakehouse**, unifying **big data operations**, **analytics workloads** & **ML workflows**.
- Fine-tuned Spark jobs on **AWS EMR** to achieve optimal performance, reducing processing times by 30%.
- Architected scalable **data models**, including **star** & **snowflake schemas**, to enhance **data storage & query efficiency**.
- Implemented **Databricks Lakehouse** to integrate **data engineering pipelines** with analytics & AI model training.
- Ensured **high data quality & accuracy** using automated validation frameworks integrated with **Spark workflows**.
- Partnered with stakeholders to design & deploy scalable **data solutions** for business-critical reporting & analytics.
- Designed **real-time ingestion pipelines** using **Apache Kafka & Spark Streaming** for near-instant data availability.
- Automated deployment processes by creating **CI/CD pipelines** with tools like **Jenkins & GitLab**.
- Used **Databricks Lakehouse** to optimize big data workflows, combining analytics, machine learning & reporting capabilities.
- Successfully migrated **on-premises databases** to cloud solutions like **AWS RDS & Redshift**, ensuring seamless transitions.
- Worked closely with **DevOps teams** to implement **performance monitoring & infrastructure optimization** using AWS CloudWatch.
- Utilized **AWS Lambda** for **severless** data processing, **reducing infrastructure costs** by 34% and also **improving scalability.**
- Designed and implemented **real-time data pipelines** using Kafka, Spark, and Hadoop to process high-velocity data streams for 1.3 Billion Records Per Hour High Performance - End to End Data Engineering Project.
- Reduced the time needed to provision cloud resources by 70%, decreasing deployment times from days to hours by developing reusable **Terraform** modules and implementing **CI/CD pipelines** for infrastructure deployment.
- Created and optimized **ETL pipelines** using **Azure Data Factory**, leading to a 20% increase in data processing **efficiency** and enabling real-time data insights
- Designed and implemented a robust data lake using **GCP's BigQuery** and Cloud Storage, improving data accessibility and reducing data retrieval time by 25%. Also, **optimized SQL queries** on BigQuery, resulting in a 40% decrease in query execution time.
- Architected and implemented a robust **data governance** strategy using **Azure Purview**, improving **data quality** and **compliance** across the organization.
- Developed and deployed **machine learning models** using **AWS SageMaker**, improving **predictive analytics** accuracy by 25% and driving data-driven decision making.
- Developed **ETL pipelines** for ingesting and transforming data from diverse sources into **Redshift**, leveraging AWS **Glue** and **Lambda** for automation.
- Collaborated with **cross-functional teams** to understand data needs, **researched** and developed custom data solutions, resulting in a 15% **increase** in **operational efficiency**.
- **Migrated** on-premise systems to **Azure**, leveraging **Terraform** and **CI/CD** for automated provisioning.
- Developed CI/CD pipelines using **Docker** and **Git Action**, ensuring efficient deployment across environments.
- Integrated A **Models into production** after converting to ONNX and using endpoints from AWS and GCP Vertex endpoints

- Improved system monitoring and reliability using **Prometheus**, Grafana, and **AWS CloudWatch**, ensuring SLA compliance for data platforms.
- Spearheaded and implemented a scalable data processing system that improved **data ingestion speed** by 30%, leading to faster insights and decision-making.
- Designed and implemented **a knowledge graph** using Neo4j, improving recommendation accuracy by 15% in ML models
- Automated **data quality checks** using Great Expectations, increasing data integrity by 98% and reducing manual data validation efforts by 20 hours per week.
- Designed and implemented a **serverless ETL workflow** with AWS Glue, reducing infrastructure maintenance costs by 40%.
- **Optimized data storage strategy** for a high-traffic e-commerce platform, reducing query costs by 50% and improving query speed by 30%.
- Built scalable **MLOps pipelines** to deploy machine learning models for real-time fraud detection in cloud environments.
- Supervised and mentored a team of associate data engineers, promoting a dynamic environment of continuous improvement, yielding a 22% increase in team efficiency.

DATA ENGINEER & MACHINE LEARNING ENGINEER

Kroger | Cincinnati, OH | **Nov 2016 – Jul 2019**

- **Developed real-time data pipelines** using Apache Kafka, Spark Streaming, and AWS Kinesis, capable of processing **10 million events per second**, ensuring high data availability and reducing downtime by 30%.
- Designed and maintained **data warehouse solutions** using Snowflake, optimizing query performance by 40% through clustering and materialized views.
- **Implemented a data lake architecture on AWS** (S3, Glue, and Athena) to centralize data storage, reducing data retrieval latency by 50% and saving **$200,000 annually** in infrastructure costs.
- Designed and implemented robust **authentication** and **authorization mechanisms**, ensuring secure access control and safeguarding **sensitive data across applications**.
- Built and maintained **continuous integration/continuous delivery (CI/CD) pipelines** using **Github Actions**, streamlining the development, testing, and deployment processes.
- Leveraged **AWS** and **Azure service operators** to streamline continuous deployment (CD) from **Github** to **Kubernetes**, enhancing application deployment efficiency.
- Created a **distributed ETL pipeline** using Airflow and Apache Spark to process **500 GB of data daily**, improving job execution times by 45%.
- Documented **REST APIs** using **Swagger** for clear communication.
- **Increased database connectivity** by utilizing **Entity Framework** and **LINQ** to Entities to connect to databases and perform various database transactions, ensuring efficient data management.
- Employed **XSD** and **XSLT** to transform and validate **XML data**, enhancing data interchange and ensuring compatibility across systems for **Data Transformation** and **Validation**.
- Implemented **RESTful APIs** and **GraphQL** to communicate with content management systems and **data pipelines**, enabling data retrieval and manipulation.
- Enhanced testing and reliability by writing **automated tests at the unit and environment levels**.
- Utilized **state-of-the-art frontend optimization techniques**, including code splitting, lazy loading, and image optimization, to improve page load times, enhance user experiences, and boost search engine rankings.
- Spearheaded the transition from Angular to Blazor for a suite of enterprise-level web applications, enhancing performance and maintainability while ensuring seamless integration with existing systems.

- Addressed critical **memory leaks** in the system, optimizing page load times by transitioning from **SASS to styled-components**, effectively managing resource utilization.
- Implemented and customized **MS CRM Dynamics solutions**, including workflows, plugins, and custom entities, to meet specific business needs.
- **Analyzed performance** by conducting comprehensive **unit and integration testing** of Angular components using testing frameworks such as Jest, Jasmine, and Karma.
- Analyzed **JMeter test results** to identify bottlenecks and optimize application performance.

MACHINE LEARNING ENGINEER & DATA ANALYST

Intel Corporation | Los Angeles, CA | **Jan 2014 – Jun 2016**

- Worked in **migration of various back-end PHP systems to C# services**, transforming a significant monolithic application into a **microservices architecture**, emphasizing expertise in .NET and Java stack.
- Proficiently **clustered MongoDB** with ~20GB live data into **PostgreSQL** using **Kafka**.
- Demonstrated advanced **DevOps skills** with **container orchestrations** like Kubernetes and managed CI/CD operations in **Jenkins** and **Gitlab**, streamlining software deployment in Microsoft ecosystem.
- Employed **Angular** for web-based client-side interfaces and integrated secure **real-time communication** with **Django** and **WebSocket** technologies on the backend, illustrating full-stack expertise.
- Integrated **Elasticsearch**, **Postgres**, and **PostGIS** to develop a **location-based search engine**.
- Developed **REST APIs** using **Python Django Rest Framework** and **MongoDB** for robust data management.
- Configured **uWSGI** to handle high connection loads, enabling real-time communication between a Python web server and the JavaScript-based frontend, which aligns with full-stack development.
- Utilized **Scrapy** to extract data from **social platforms**, showcasing **data handling skills** within the Microsoft stack.
- Enhanced the **frontend development process**, including coding guidelines and code quality tools, promoting best practices for JavaScript in frontend development within the **Microsoft technology stack**.
- Employed **Node.js** and **Redux** to interact with **DatoCMS** for **Headless CMS**, emphasizing full-stack development with Microsoft technologies for API integration.
- Designed, built, and conducted **unit testing of software** in a collaborative environment, showcasing teamwork and software development skills within the Microsoft technology ecosystem.