QIANYI SHA

+19493830343 | shhh9712@gmail.com | Seattle, WA, USA | linkedin.com/in/qsha/

SKILLS

Languages: Python, JavaScript, Typescript, Java, SQL, HTML/CSS, Go, C/C++
Frameworks: React.js, Next.js, Vue.js, Node.js, Redux.js, AngularJS, Django, Express.js, Jest, Spring, Flask
Data: Oracle, S3, Postgres, Apache Spark, Airflow, Apache Kafka, AWS, BigQuery, Snowflake, Databricks, bigquery, informatica, Kafka, Event-hub, Azure, AWS, SQL, NoSQL, DBT, Glue, Lambda, Azure Blob Storage, Azure Data Factory, GCP
Other: CircleCI, github, CI/CD, gRPC, Protobuf, kubernates, Linux/Unix, Docker, Computer Networking, Computer Vision, Data Structures & Algorithms, OOP, Unit Testing, FrontEnd, Backend, Full-Stack, Microservices, Distributed System, MVC

PROFESSIONAL EXPERIENCE

Ascend.io **Menlo Park, CA, USA**
Field Data Engineer *October 2022 - March 2024*

- Maintained and developed backend micro-services written in Go, Python, GRPC, Google Protobuf supporting functionalities like connector framework Spark worker management ORM among others
- Maintained and developed front-end interfaces utilizing React, JavaScript, Redux, Node.js, Bootstrap amon others.
- Contributed to developing multiple frameworks for connecting different types of databases (MySQL Oracle Postgres) including REST API S3 Dynamo RedShift Azure Event-hub Google Analytics etc., enabling batch/streaming/CDC ingestion methods,
- Developed and enhanced an internal billing pipeline that processes 50 million rows of billable event data daily. This supports the company's business needs and provides insights into $300M in revenue. The process involves ingesting data from Google BigQuery and Loki log parser, transforming it with PySpark and SQL, and generating BI reports using Looker and Dash.
- Ensured site reliability by managing infrastructure components such as Kubernetes clusters, Docker, Nginx, Cilium, and MySQL. Additionally, provided on-call support to address production issues. Utilized a cluster monitoring stack comprising Grafana, Loki, Hubble to troubleshoot issues in Linux pods.
- Optimized data pipelines by employing reusable PySpark clusters to improve compute cluster utilization rates, enhancing storage flexibility through AWS and Azure Blob Storage with Iceberg, and boosting pipeline runtime efficiency with PyArrow. These optimizations resulted in a 30% average cost reduction and an 82% decrease in daily processing time from six hours to just over one hour.
- Successfully engineered and deployed an end-to-end service for the automatic migration of 1200+ data pipelines across 10+ enterprise companies to a new platform, utilizing Informatica, DBT, and Airflow. Facilitated seamless transitions across various platforms and also saved 1000+ engineering hours of manual migration work.
- Designed CI/CD process specifically for deploying data pipelines efficiently while automating tests using CircleCI. A Python CLI was developed to streamline all related operations ensuring smooth running on production systems.
- Created ETL pipelines that work with cloud warehouses like Snowflake, Databricks, Bigquery; leveraging features such as Snowpark and Unity Catalog.
- Proposed guidelines for internal incident triage and handling as a preventative measure against recurring incidents.
- Integrated systems for the customer success department by connecting Zendesk, Slack, Intercom, Planhat through internal system integrations.

Youyuan **Beijing, China**
Software and Data Engineer Intern *January 2021 - July 2021*

- Design and implement data collection points along with mechanisms for gathering data. Create a preprocessing workflow to ensure the high quality of data before it enters subsequent stages.
- Combine Machine Learning models, including PCA and XGBoost, trained on historical user activity. Establish a robust training and inference cycle at production level. Achieved an R-square of 0.92, significantly improving A/B testing efficiency by reducing test times by 30%.
  * Utilize Python and Pandas for data cleaning, feature engineering, and normalization
  * Apply dimensionality reduction through PCA using scikit-learn
  * Split datasets for XGBoost model training with ScikitLearn; perform cross-validation
  * Evaluate model performance using MAE, RMSE, MAPE, and R-square metrics
  * Implement a batch processing pipeline to collect new data continuously; clean it up; process it through PCA before evaluation by the model
  * Track model performance via metrics collected by a Python agent

EDUCATION

University of Washington **October 2021 - December 2023**
Master's, Computer Engineering *GPA: 3.87*

University of California - Irvine **October 2016 - December 2020**
Bachelor's, Computer Engineering