Drew Yang

214-727-9608 | drew.yang.dev@gmail.com | Houston, TX | LinkedIn | Github | Website

Technical Skills

- **Code:** Python, bash, SQL, Markdown, powershell, Java, C/C++
- **CI/CD:** Github Actions, Azure DevOps, Jenkins, ArgoCD
- **Cloud/IaC:** AWS, Azure, Terraform, Packer, Ansible
- **Config:** cloud-init, SaltStack
- **Container:** Docker, Podman, Kubernetes(kOps, kubectl, helm)
- **Observability:** OpenTelemetry, CloudWatch, Datadog, Grafana

Experience

Software Engineer(DevOps) **July 2021 - Present**

DataJoint - Science Operation for Neuroscience. [Open-source] and [Commercial] *Houston, TX*

* ***AWS:*** Administrated DataJoint's AWS account and several other customers' AWS accounts. Configured **VPC**, **Subnet**, **Security Groups**, **IAM** role and policies, **S3** lifecycle management, **EFS** access point, **EC2** instances, **RDS** instances, **Lambda** triggered by **SQS** or **EventBridge**, **SNS** and **SES**, **CloudWatch** metrics and alarms, **Route 53** DNS records, **Secrets Manager** for deployment secrets.
* ***CI/CD:*** Developed generic **Github Actions** reusable workflows used by **30+** repositories followed by Conventional Commits, Release Flow and GitOps best practices, to automate build, test, release, publish private or open-source **Python** packages[PyPI] or deploy **Docker** images[Dockerhub].
* ***Kubernetes:*** Provisioned Kubernetes clusters hosted on EC2 instances for development, staging and production environments using **k3d or kOps**. Developed utility **bash** scripts with **helm** and **kubectl** to manage Kubernetes clusters more efficiently, including configuring **Nginx ingress** controller, cert manager with **Let's encrypt** issuer, **Cillium** Container Network Interface(CNI), IAM Roles for Service Account(**IRSA**), **Cluster Autoscaler**, AWS Elastic Load Balancer(**ELB**) or deploying applications like Percona XtraDB Clusters, Keycloak, JupyterHub, Flask and ReactJS based web application, etc.
* ***Ephemeral Worker Clusters:*** Designed and developed a worker lifecycle manager using Python within one month to fulfill an **urgent** business requirement. This development **polls** jobs from a MySQL database, then provisions and configures ephemeral EC2 instances by **Packer(pre-build AMI), Terraform and cloud-init** to compute jobs **at scale**; implemented AWS S3 mount to significantly reduce raw data downloading **overhead** and added EFS as a file cache for intermediate steps to improve computation **failover**; configured **NVIDIA CUDA toolkit** and **NVIDIA container runtime** for **GPU** workers.
* ***Platform Automation:*** To provision or terminate AWS resources using **boto3** or **Terraform**; manage customers' **RBAC** permissions using Keycloak and Github REST API; generating usage and billing report with **AWS S3 Inventory** report, **AWS CloudTrail** and **AWS Cost and Usage** report, made a **Plotly Dash** to analyze cost and usage efficiency.
* ***Jupyterhub:*** Configured and maintained Jupyterhub deployment on a Kubernetes cluster with **Node Affinity** to assign pods onto different nodes by requirements and **Cluster Autoscaler** along with **AWS Auto Scaling Group** to accommodate **100+** active users; improved base images' **build time** and maintenance **overhead**.
* ***Observability:*** Implemented a small part of the metrics and alerts using **AWS CloudWatch**, and then later integrated **Datadog** for Kubernetes clusters' and ephemeral EC2 instances' metrics and logging through **OpenTelemetry** protocol, synthetic API testing, and UI/UX monitoring.
* ***Security:*** Set up codebase **vulnerability** scan with FOSSA; Set up **AWS Secrets Manager** working with **External Secret Store Operator** to secure Kubernetes secrets; Deployed and administrated self-hosted **Keycloak** for **RABC** authentication, further integrated it with **AWS IAM** as an **identity provider** to access AWS resources through **STS**, enabled OpenID Connect(**OIDC**) authentication flows such as authorization code flow, client credential flow, password grant flow etc.
* ***MySQL Database:*** Maintained a self-hosted **Percona XtraDB Clusters** on database **daily backup** stored on **S3**, **mysqldump** backup redundancy, Point-in-Time Recovery(**PITR**), **deadlock** detection, and slow query log.

Software Engineer(MLOps) **May 2019 - July 2021**

dataVediK - Optimize Oil & Gas operations by Machine Learning. [DrillVedik] *Houston, TX*

* ***Interactive Drilling Dashboard:*** This is an **enterprise** product that I worked with two more engineers. Developed a **Plotly Dash** dashboard that visualizes processed data using Bootstrap, CSS media query, **Redis** and sqlalchemy. Also, implemented a **socket** service will notify when **Airflow** pipeline finished processing in order to **synchronize**(refresh) the dashboard's data.
* ***CI/CD Pipeline:*** Set up several **Azure Pipelines** for continuous development, testing and continuous deployment in **dev, test**

and prod stages. Additionally, made a **Jenkins** pipeline to work with on-premise infrastructures.
**** ML Pipeline:*** Set up a **MLflow** server for machine learning experiment logging, parameter tuning, continuous training, model management and model serving.
**** ETL Pipeline:*** Working with a data engineer, set up an **Airflow** server for our data ETL pipeline.
**** Prediction Task Manager:*** Working with a front-end developer, designed and developed a **production** web application that supports job queuing and parallel processing for drilling speed prediction using JavaScript, **flask**, sqlalchemy, **celery**, RabbitMQ, gunicorn, Nginx, supervisord, Docker and AWS EC2, AWS Cognito Authentication, HTTPS
**** Drilling Status Detection:*** Working with a domain expert, developed two **classification** models for detecting drilling status using Logistic Regression and Random Forest with the convenience of the MLflow server
**** Drilling Speed Prediction:*** Working with a domain expert, applied Gaussian Process **Regression** for feature synthesis based on geographical information as well as **feature engineering** based on correlation matrix and F1 score ranking, built a non-linear regression model using LSTM RNN.

Education

Southern Methodist University, *Master's of Computer Science* Dallas, TX | Aug 2017 - May 2019

Qingdao University, *Bachelor's of Software Engineering* Qingdao, China | Aug 2013 - May 2017