Dean Kuhn

Milwaukee, WI Area | Open to Remote | dakuhn1138@gmail.com | LinkedIn | GitHub

SUMMARY

Computer science graduate with a focus on data engineering and pipeline development. Built four end-to-end portfolio projects spanning automated data pipelines, ML-driven forecasting, and algorithm optimization — including two pipelines on scheduled GitHub Actions, a full Postgres/Snowflake/dbt analytics stack, and a configurable Genetic Algorithm optimizer. Eager to apply hands-on engineering skills in a junior data engineering, analytics engineering, or software development role.

PROJECTS

KitchenSync Food Forecasting System *2026*

Tech: *Python · FastAPI · PostgreSQL (Neon) · Snowflake · dbt Core · LightGBM · scikit-learn · Streamlit*

- Architected a full end-to-end forecasting system simulating a retail kitchen production pipeline: async FastAPI ingest layer, per-store schema isolation across 12 Neon Postgres schemas, Snowflake analytics warehouse, LightGBM forecasting model, and a Streamlit dashboard with 60-second refresh showing production queues and missed demand from stockouts.
- Built a three-layer dbt Core pipeline (staging → intermediate → marts) in Snowflake across 6 mart models: production targets, item velocity with urgency flags, waste percentage by category, cold-start profiles, stockout summaries, and store-level sales aggregates.
- Engineered an async POS simulator (httpx + asyncio) using Poisson customer arrivals and a time-of-day rush curve; each store runs as an independent async task with a StoreState class tracking FIFO batch inventory, expiration, and stockout events.
- Implemented cold-start handling for items with fewer than 4 data points (falls back to category-level mart averages) and a configurable urgency threshold (URGENT when sell-through exceeds 2x historical average); orchestrated full pipeline via a single run_pipeline.py script (extract → dbt → train → predict).

Music Growth Pipeline *2026*

Tech: *Python · PostgreSQL (Neon) · dbt (dbt-postgres) · Last.fm API · GitHub Actions · SQL*

- Designed and deployed a weekly ingestion pipeline that snapshots listener and playcount data for 7,755 artists from the Last.fm API into a cloud Postgres database — building longitudinal time-series data for engagement analysis, a capability the API does not natively provide.
- Built a dbt transformation layer (staging + mart models) on top of raw Postgres tables: stg_* models clean and rename source data; mart models artist_tiers, genre_stats, and artist_similarity_network power cross-sectional and network analysis.
- Extended the schema to capture genre associations (15 genres × 500 artists) and artist similarity networks (~2,000 indie artists, 20 similar artists each), enabling genre-level breakdowns and cross-tier network analysis.
- Wrote cross-sectional SQL analysis surfacing a consistent ~4x plays-per-listener gap (mainstream median 74.76 vs. indie 17.69) across the full distribution; automated weekly snapshots via GitHub Actions (Sunday 9AM UTC).

Market Cynic Pipeline *2026*

Tech: *Python · Playwright · VADER (NLTK) · Pydantic · pandas · PyArrow · Streamlit · GitHub Actions*

- Built a Bronze→Silver→Gold medallion pipeline correlating Reddit sentiment (4 subreddits) with Yahoo Finance price data to detect sentiment-price divergence — a signal for potential retail investor mismatch.
- Engineered "The Cynic Heuristic": a two-layer weighting system combining a per-post controversy signal weight (log-scaled by comment count) with per-subreddit trust multipliers, producing a single normalized sentiment score per ticker.
- Implemented rolling divergence detection (6-run window) comparing sentiment momentum to price momentum; results surface in a live Streamlit dashboard with dual-axis charts and flagged divergence events.

Package Delivery Routing System (WGU Capstone) *2026*

Tech: *Python · Genetic Algorithm · CLI*

- Replaced a static nearest-neighbor routing algorithm with a custom Genetic Algorithm that simultaneously optimizes package-to-truck assignment and delivery order — a combined loading and routing problem handled by a single fitness function.
- Implemented adaptive mutation (rate doubles after 50 stagnant generations, resets on improvement), scramble and inversion mutation operators, and sentinel-aware ordered crossover to escape local optima and maintain constraint validity.
- Delivered as a configurable CLI tool supporting custom package counts, truck capacity, refrigeration constraints, and GA hyperparameters; includes per-package status lookup by ID or address post-run.

TECHNICAL SKILLS

Languages: Python, SQL
Data & Storage: PostgreSQL, Snowflake, dbt Core (dbt-postgres, dbt-snowflake), pandas, PyArrow, Parquet
ML & Modeling: LightGBM, scikit-learn, VADER sentiment analysis (learning)
APIs & Ingestion: REST APIs (Last.fm), FastAPI, Playwright (headless scraping), Reddit public feeds
Tools & Platforms: GitHub Actions, Neon (cloud Postgres), Streamlit, Git, WSL, Linux, AWS (exposure)
Concepts: Pipeline design (medallion architecture), data modeling, dbt staging/mart/metrics patterns, ML inference serving, genetic algorithms

EDUCATION

B.S. Computer Science · Western Governors University *Expected May 2026*

WORK EXPERIENCE

Guest Services Coworker · Kwik Trip *2021 – Present*

- Maintained consistent reliability and performance over 4+ years in a high-volume, customer-facing environment.
- Handled cash transactions, inventory, and shift responsibilities; recognized for dependability across extended tenure.