

## David Lohmann

<https://www.linkedin.com/in/DavidLohmanniii/>

<https://github.com/DLohmann>

<https://www.kaggle.com/dlohmann>

<https://dlohmann.me/>

### Recruiting Summary:

- David is a highly skilled and experienced software engineer with a strong track record of success.
- With over 3 years of experience at Google, David has demonstrated expertise in Python, C++, Java, SQL and data processing pipelines.
- Their experience at Google, specifically in the Geo and Play Apps Search Quality teams, demonstrates a strong foundation in the design, development, testing, and evaluation of complex systems and software, which is central to the role.
- David's work on migrating Google Maps's People Flow services from monolithic to microservice architecture, and their involvement in building infrastructure to serve and render vector map data, showcases their ability to modify existing systems and contribute to the development of data-intensive systems. This experience directly correlates with the job responsibility of modifying existing data and client systems and working as part of a project team.
- Moreover, David's proficiency in a wide range of programming languages and technologies, including advanced skills in C++ and Python, SQL and data processing pipelines aligns with the job's requirement for knowledge of computer development software. Their hands-on experience with data processing pipelines, unit testing, and code reviews further underscores their capability to fulfill the job responsibilities outlined.

### Skills Matrix By Recruiter:

Skills	Total Years of experience	Proficiency
C++ programming	3+ years	Expert
Python Programming	3+ Years	Expert
Large scale data processing pipelines	3+ years	Expert
SQL	3+ years	Advance - Expert
Apache Beam (Flume)	2+ Years	Expert

### Summary:

- David has expertise in research and software development which include Cloud Computing, DevOps, Data Science, Artificial Intelligence, Machine Learning, Numerical Methods, Optimization, and Computer Hardware.
- Skilled in C++, Python, C, Java, SQL and data processing pipelines.

### Software Skills:

- **Advanced:** C++, Python
- **Proficient:** SQL
- **Moderate:** CUDA, JSON, MATLAB
- **Tech/Libraries:** NumPy, SciPy, matplotlib, pandas, BigQuery, Apache Beam, Protocol Buffers
- Bash Linux Terminal, Git, Docker, SSH and Jupyter/Colab/VsCode/Kaggle notebook
- Azure MLOps Challenge

### Engineering Skills:

- Data Processing pipelines
- Unit testing, Code reviews
- Design documentation
- User metrics, Statistics, Math
- Geographic Information Systems. ArcMap (ESRI), and Google Maps. Spatial Analysis, and Modeling (GIS), Geographic data
- Built dlohmann.me with Azure Static Web App and Bootstrap

### Professional Experience:

**Meta, Remote**

**Jun 2024 – Dec 2024**

**Contractor through Mindlance, Software Engineer**

**Project: Ads Fairness Equity, Variance Reduction System**

**2024**

- Worked to ensure advertising for housing, employment, or credit ads are fair and do not discriminate for or against any gender, age, or estimated race. Ensured that neither advertiser preferences nor Meta's ad algorithms do not discriminate against any protected category. Ensured that Meta's Variance Reduction System requires machine learning ad algorithms to conform to legally specified fairness metrics. Reduced Meta's legal risk in the US and CA pertaining to discriminatory housing, employment or credit ads served by gender, age, or race. Contracted through Mindlance.
- Built and maintained data pipelines, infrastructure, dashboards, and data visualization tools to allow Meta's leadership to easily understand the extent that Meta's ad platform conforms to anti-discrimination legal requirements.
- Translated legal and technical documentation into data infrastructure to calculate various machine learning fairness metrics and display these metrics and utilizing differential privacy. Supported work on machine learning efforts to ensure that Facebook's Ad Creatives product generative AI tool does not generate discriminatory ads.
- Used SQL, Python, Notebooks, Apache Hive, Dataswarm, Presto, and technical documentation.

**Professional Experience:**

**Google, Mountain View, CA**

**Jul 2019 – Nov 2022**

**Software Engineer**

**Project: Geo Extended Maps & Tiling:**

**2022**

- Worked on building infrastructure to serve and render vector map data on maps clients (web and mobile) using data from server backends of other teams, and specifically the pipeline and server to serve these render operations.
- Worked on a pipeline to process map data and validate whether it is correct before storing the data so it can be accessed and served by the server.
- Added rules to check whether map features and pipeline counters are correct and have changed drastically between pipeline runs.
- Migrated code to use memory arenas (also called "regions") for faster and more efficient processing of protocol buffers when building labels. Refactored server code and added more unit tests.
- Added logic to ensure small postal codes appear at certain zoom levels.

**Google - Geo People Follow & Place Follow:**

**2020 - 2022**

- Migrated Google Maps's People Follow services from monolithic architecture to microservice architecture.
- Worked on migrating the backend endpoints serving tactile (web) and mobile (iOS and Android)
- People Follow services on Google maps from using Google Web Service To Boq (Google's microservice architecture).
- Added onto Mapsfe, Google Maps' boq implementation.
- Worked with a team in Tokyo that completed the migration.
- Assisted the Tokyo team in taking over the project, and helped make some design docs and present the system to the new team.
- Also worked on following for places, and made algorithmic improvements to infrastructure using stubby to sync changes to followed place lists, increasing efficiency.
- This task required learning Apache Beam (Flume), Stubby, gRPC, Protocol Buffers, Boq, GWS, Google Apps Framework, Producers, Guice, Java.

**Google - Geo Indoor Maps:**

**2020**

- Worked on auto-generating indoor maps based on location data.
- The goal of this project is to use aggregated location trace data to replace the slow surveying process normally used to generate indoor maps (with the help of human operators) with a fast, automatic, and scalable way to generate indoor maps and related point of interest relations.

- The goal was to eventually put these changes into Google Maps' databases, thus benefitting Geo's ecosystem and improving user experience, particularly in commonly visited buildings such as shopping malls, while avoiding private areas.
- Used Google Colab, Python, and SQL for data visualization. Wrote Apache Beam (Flume) C++ pipelines to process indoor walkway virtual segments, and unit tested these pipelines.
- Processed virtual segments to get intersections of indoor walkways.
- Generated edits so these intersections can be added to Google's databases, and ensured that these new walkways were registered as being within certain buildings.
- Worked on a feature to refresh indoor walkways and remove stale walkways from Google Maps' databases.
- Created a metric for the indoor walkway refresh pipeline to determine whether a building's walkways are up to date, by classifying walkway changes as large-scale rebuilding, or small remodels.
- Wrote a design summary for most work done so far on the auto-generated indoor maps team.

### Google - Play Apps Search Quality:

2019 - 2020

- Worked on Play Apps Search's related query suggestions cluster.
- Filtered out queries for apps that are not in the play store, so they will not be suggested to users.
- Worked on a pipeline to generate follow-up queries to be recommended in the related query cluster for play store apps searches.
- Wrote a Flume C++ pipeline to determine which apps are commonly searched for, but that are not in the play store, and filtered those apps out of the "related searches" suggestions in play store search.
- Gained experience working with related queries, which are used by many teams in several product areas (ie Play, YouTube, Image Search), and writing unit tests and Flume pipelines.
- Wrote Flume pipelines to scrape play logs and query sessions to find information about how users refined their queries, and how they engaged with the related query cluster on the play store app.
- When users search and cannot find the app they are looking for, they typically refine their search terms and keep searching.
- I wrote a Flume pipeline to process the original search terms and most common follow-up search terms and process these to calculate several metrics that can help us measure how good our related searches suggestions are.
- Calculated several key metrics, and queried infrastructure to find information about the apps that these queries would have resulted in.
- Extracted signals from these apps and output this data in a file for training play's machine learning model to rank related queries in the related query cluster better.

### Projects:

#### Spatial Poisson Process:

2021

- Data Visualization Demo. This project was originally a demo that I made as an illustration for the Poisson Point Process Wikipedia page.
- The live demo demonstrates the statistical trend of randomly distributed points to occur in certain areas following a Poisson distribution.
- Demonstrates the law of large numbers and the central limit theorem in statistics.
- Made in Typescript, Javascript, and HTML, and used the Plotly plotting library (<https://github.com/DLohmann/Spatial-Poisson-Process-Demo>).

#### Bitcoin Network Analysis:

2019

- Written in Google Colab Notebook in Python and SQL. Used pandas matplotlib, SQLite, and Igraph.
- The goal was to visualize the explosive growth in popularity of Bitcoin during its first few years, by using matplotlib and Gephi to visualize the transaction network.
- Project was part of the MIST 134 Network and Data Science course at UC Merced.

#### Recipe Suggestion:

2018

- App using Bayesian Classifier & Machine Learning in Django and hosted using Amazon Web Service (<https://goo.gl/dWMqrQ>). Front end in Java and XML for Android (<https://goo.gl/sBV5pt>).

- Written at UCLA Hackathon with 2 other programmers. Reads in what cooking ingredients you have, and uses machine learning to suggest a recipe from these ingredients.
- Worked on the Bayesian classifier recipe suggestions in Python, and the front end web page and received the JSON recipes data in JavaScript.

#### **Titanic Kaggle Competition:**

**2018**

- Used Binary Logistic Regression to predict the survival of passengers on the Titanic based on their attributes with 80% accuracy.
- Worked alone in Python (<https://goo.gl/328KRn>).

#### **Eigenvalue Analysis Tool:**

**2017**

- Utility to solve/plot phase portraits of 2 by 2 first order systems of differential equations.
- Project in Python 3.5's Numpy, Scipy, and Matplotlib libraries (<https://goo.gl/ccs9Gz>).
- Worked alone. Built for a UC Merced Applied Math Professor.

## **Education**

UC Merced 2015 - 2019

Major: Computer Science & Engineering, B.S.

Minor: Applied Mathematics

Graduation: May 2019

#### Interests

Data Science, Data Engineering, Machine Learning, Artificial Intelligence, Parallel Computing, Cloud Computing, Numerical Methods, Tensorflow, Sklearn