Software Engineering for ML/AI

Claire Le Goues

Michael Hilton

institute for SOFTWARE RESEARCH | Carnegie Mellon University School of Computer Science

Administrivia

- Homework 3 out

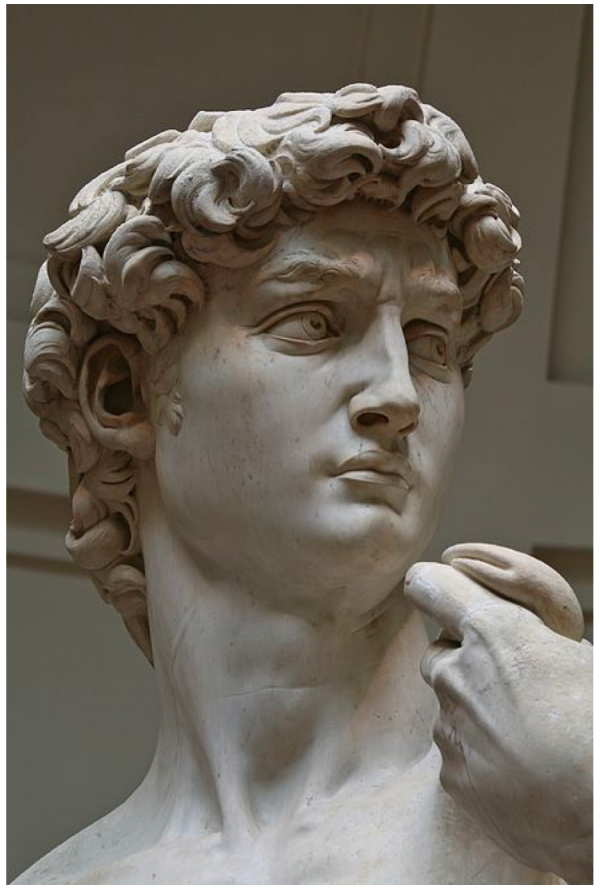
Interview: Josh Gardner

Software Engineering and ML

Traditional Software Development

"It is easy. You just chip away the stone that doesn't look like David."

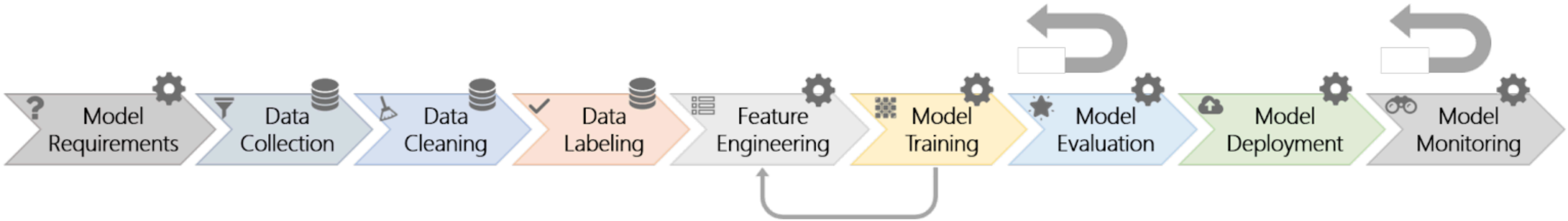
–(probably not) Michelangelo



ML Development

- Observation
- Hypothesis
- Predict
- Test
- Reject or Refine Hypothesis

Microsoft's view of Software Engineering for ML



Three Fundamental Differences:

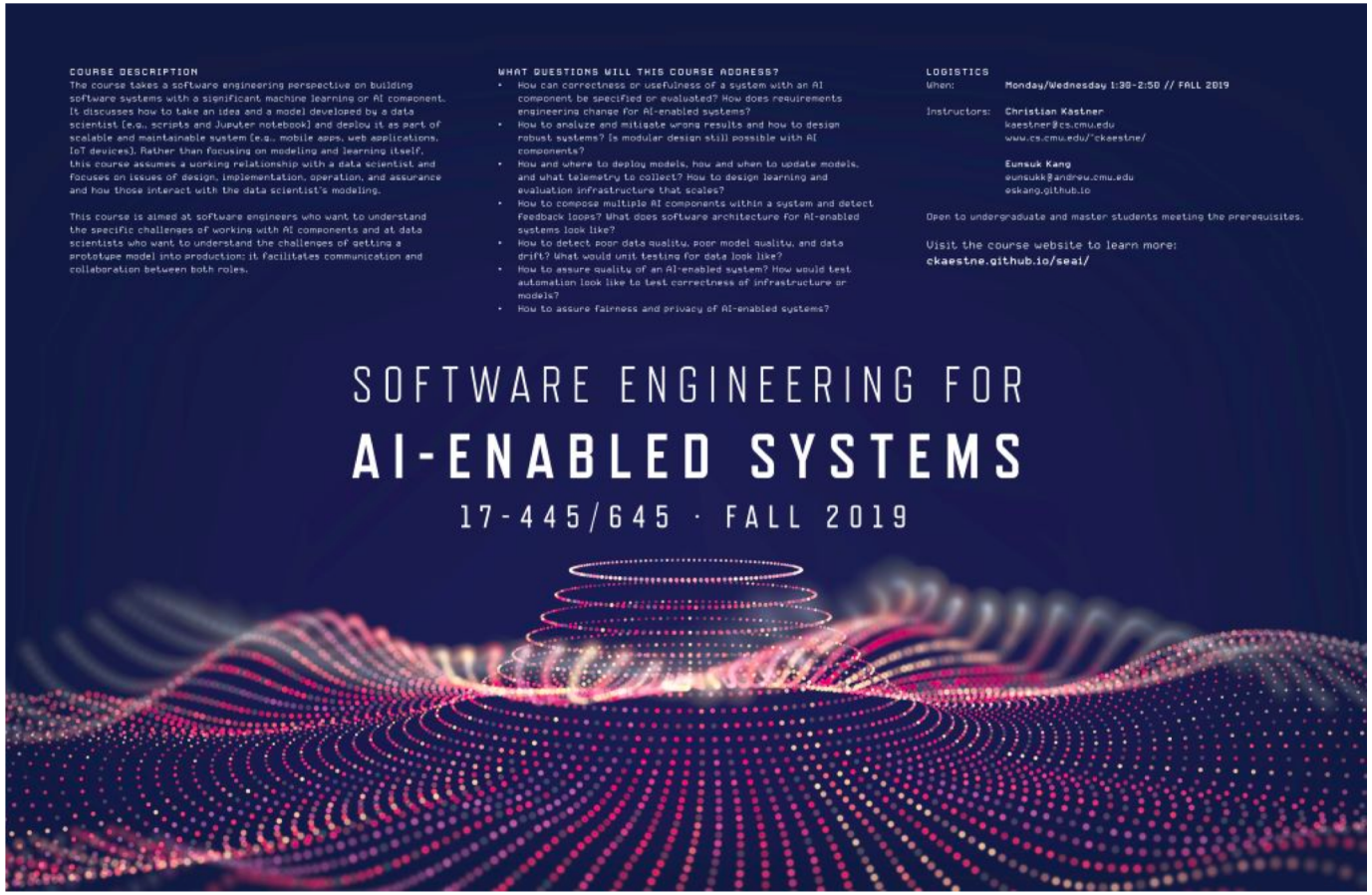
- Data discovery and management
- Customization and Reuse
- ML Modularity

Case study developed by

Christian Kästner

https://ckaestne.github.io/seai/

CASE STUDY

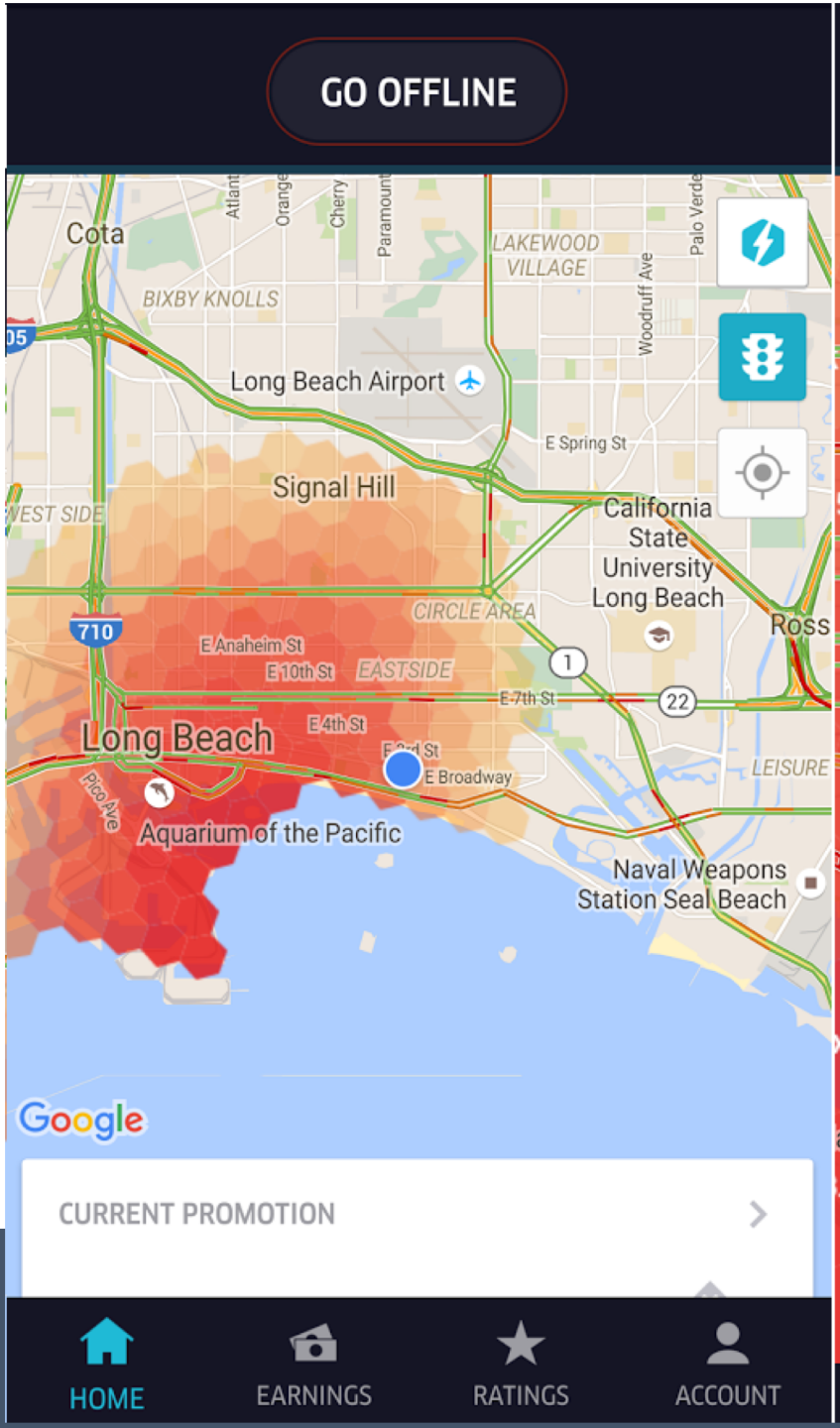


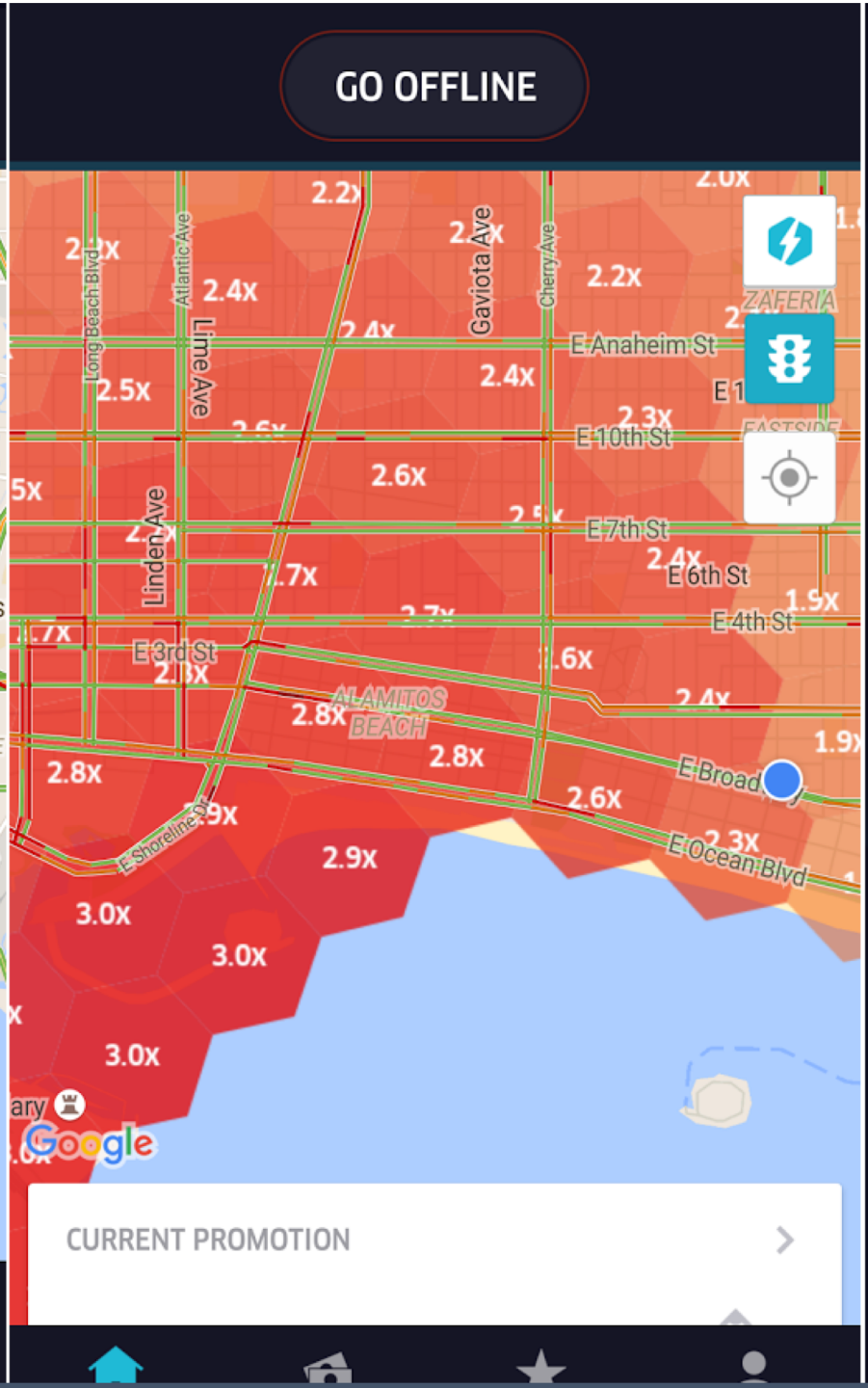
WHAT CHALLENGES ARE THERE IN BUILDING AND DEPLOYING ML?

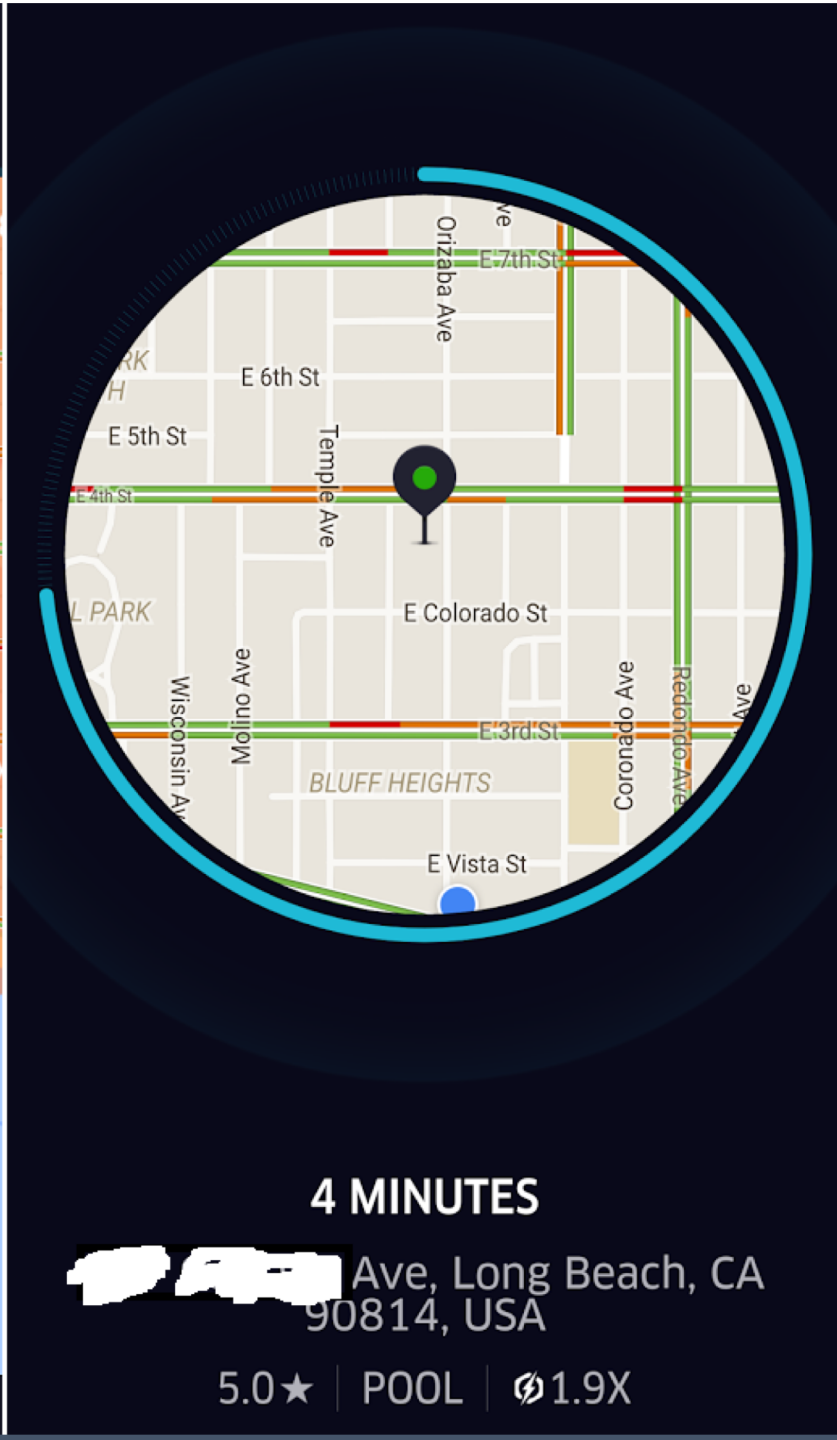
명동교자
아하 노래방
PhoDUCK
TIBETAN INDIAN NEPALI FOOD
POTALA
POTALA RESTAURANT
MINI BEER PUB
빤닭빤닭
鸡肋 タッカルビ DAKGALBI
명동본점 2F
김밥
명동교자
정성본
갈국수
B1
日本語歌完全具
六千曲收


現金のみ
Japanese English
cash only

Qualities of Interest?


GO OFFLINE
Cota
LAKEWOOD VILLAGE
BIXBY KNOLLS
Long Beach Airport
E Spring St
Signal Hill
California State University Long Beach
CIRCLE AREA
Ross
E Anaheim St
E 10th St
EASTSIDE
E 7th St
E 4th St
Long Beach
E Broadway
LEISURE
Aquarium of the Pacific
Naval Weapons Station Seal Beach
Google
CURRENT PROMOTION
HOME
EARNINGS
RATINGS
ACCOUNT


GO OFFLINE
Long Beach Blvd
Atlantic Ave
Lime Ave
Gaviota Ave
Cherry Ave
ZAFERIA
E Anaheim St
E 10th St
EASTSIDE
Linden Ave
E 7th St
E 6th St
E 4th St
E 3rd St
ALAMITOS BEACH
E Broadway
E Shoreline Dr
E Ocean Blvd
2.2x
2.4x
2.5x
2.6x
2.8x
2.9x
3.0x
Google
CURRENT PROMOTION


Orizaba Ave
E 7th St
E 6th St
E 5th St
Temple Ave
E Colorado St
Wisconsin Ave
Molino Ave
E 3rd St
BLUFF HEIGHTS
Coronado Ave
Redondo Ave
E Vista St
4 MINUTES
[redacted] Ave, Long Beach, CA 90814, USA
5.0★ | POOL | 1.9X

Qualities of Interest?

MACHINE LEARNING PIPELINE

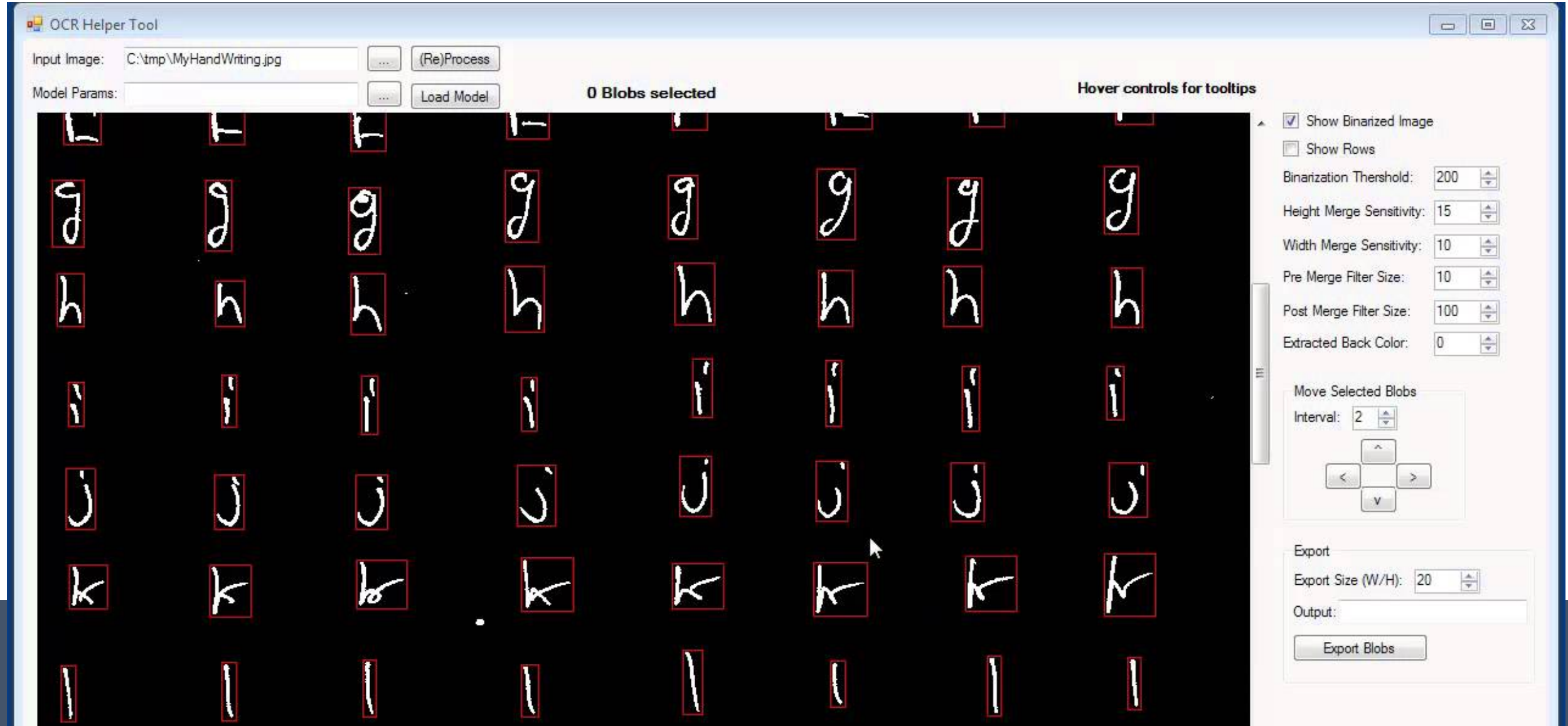
Typical ML Pipeline

- Static
  - Get labeled data
  - Identify and extract features
  - Split data into training and evaluation set
  - Learn model from training data
  - Evaluate model on evaluation data
  - Repeat, revising features
- With production data
  - Evaluate model on production data; monitor
  - Select production data for retraining
  - Update model regularly

Example Data

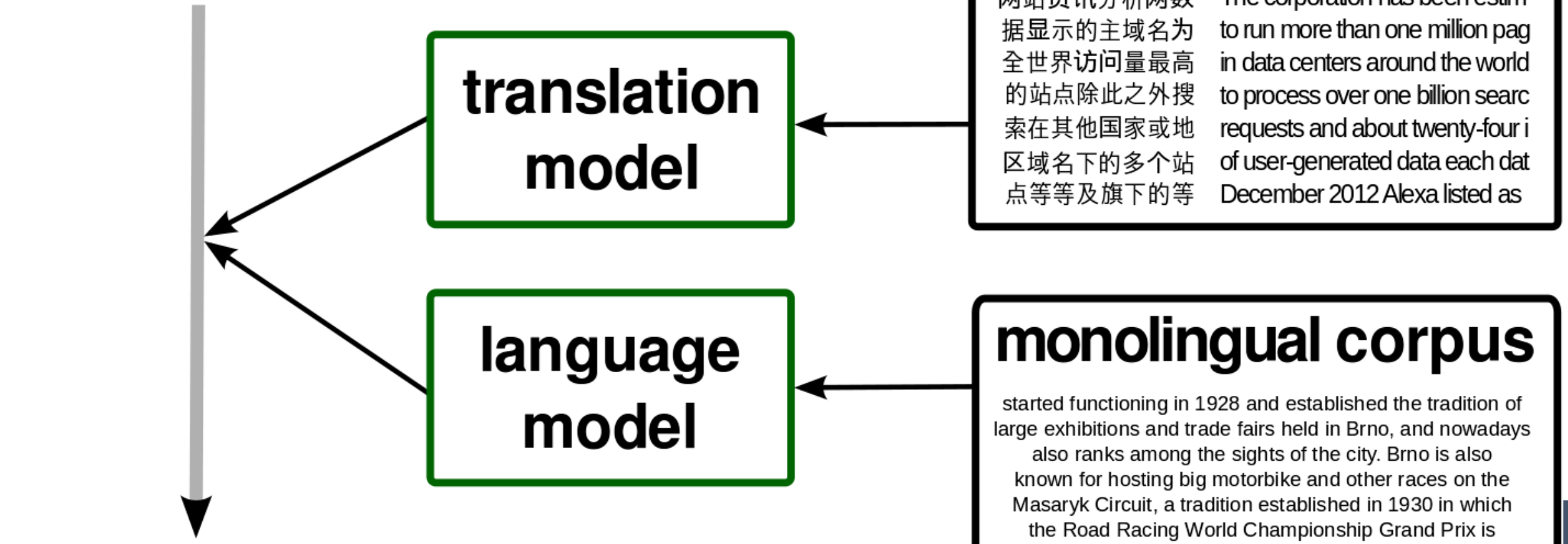
UserId	PickupLocation	TargetLocation	OrderTime	PickupTime
5	….	…	18:23	18:31
…				

Example Data



Learning Data

似乎格式有問題



English output

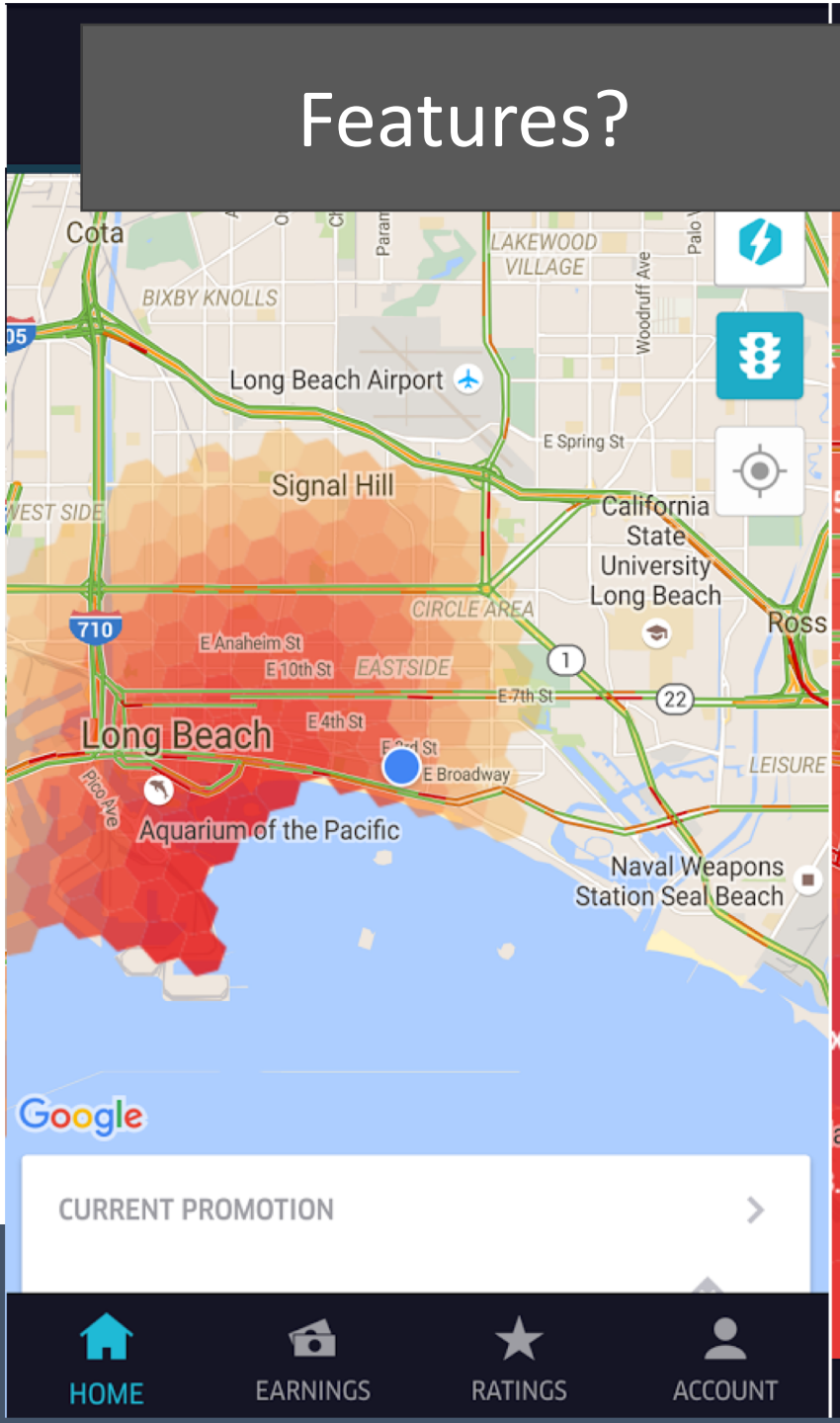
Training Data

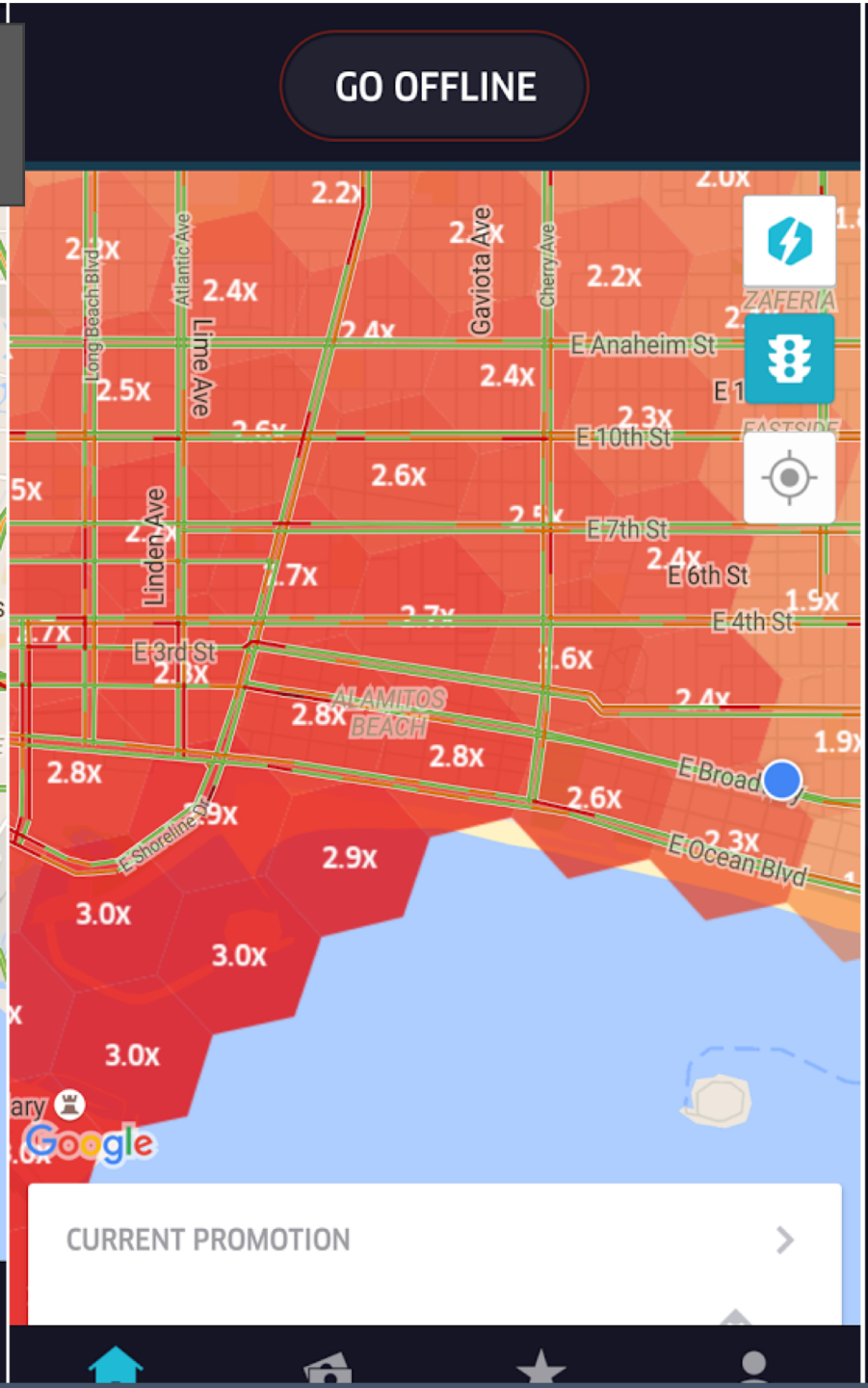
Feature Engineering

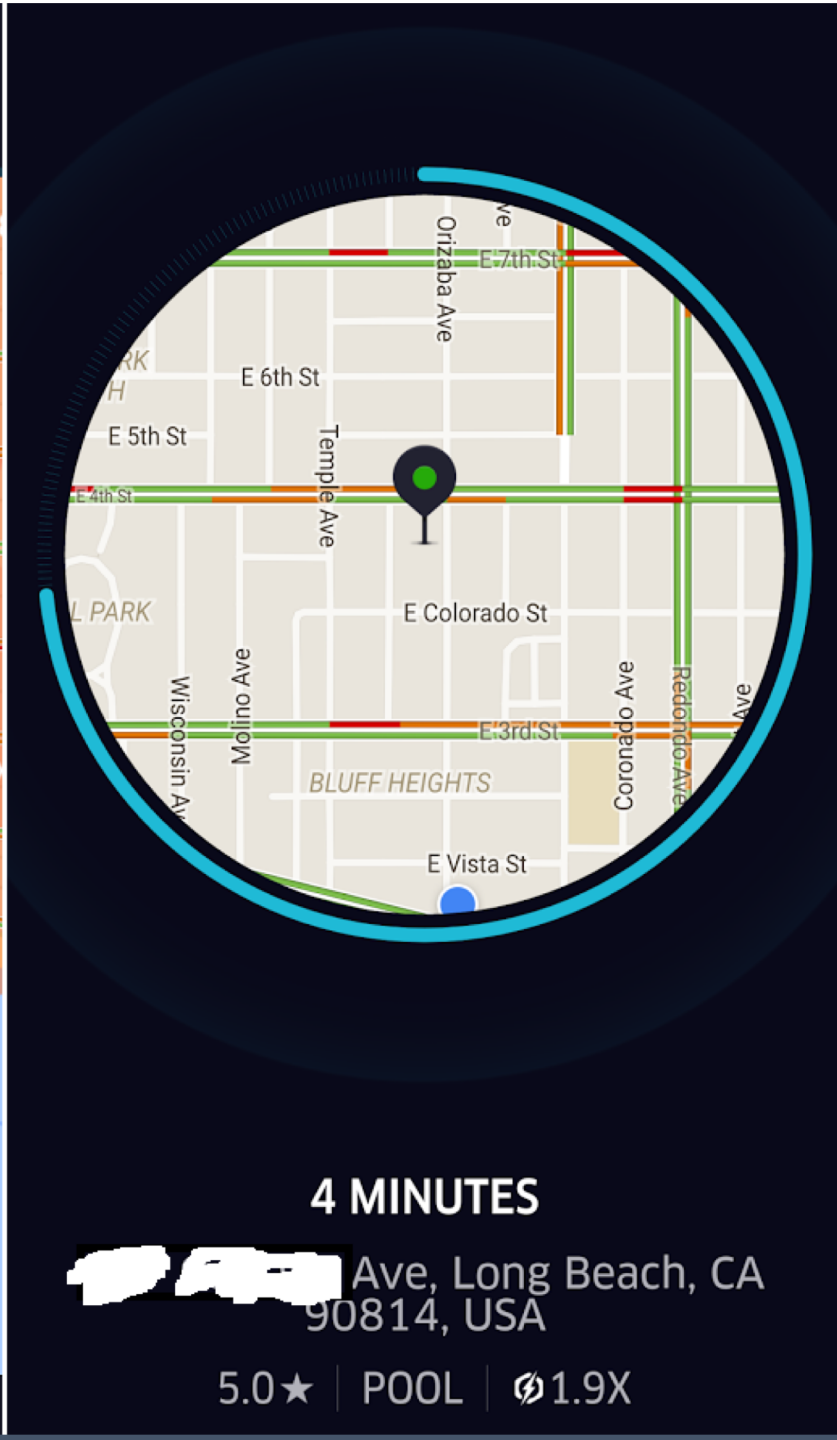
- Identify parameters of interest that a model may learn on
- Convert data into a useful form
- Normalize data
- Include context
- Remove misleading things
- In OCR/translation:

Features?

Features?


Cota
LAKEWOOD VILLAGE
BIXBY KNOLLS
Long Beach Airport
E Spring St
Signal Hill
California State University Long Beach
Ross
CIRCLE AREA
E Anaheim St
E 10th St
EASTSIDE
E 7th St
E 4th St
Long Beach
E Broadway
LEISURE
Aquarium of the Pacific
Naval Weapons Station Seal Beach
Google
CURRENT PROMOTION
HOME
EARNINGS
RATINGS
ACCOUNT


GO OFFLINE
Long Beach Blvd
Atlantic Ave
Lime Ave
Gaviota Ave
Cherry Ave
E Anaheim St
E 10th St
E 7th St
E 6th St
E 4th St
E 3rd St
Linden Ave
ALAMITOS BEACH
E Broadway
E Shoreline Dr
E Ocean Blvd
2.2x
2.4x
2.5x
2.6x
2.8x
2.9x
3.0x
1.9x
2.3x
Google
CURRENT PROMOTION


Orizaba Ave
E 7th St
E 6th St
E 5th St
Temple Ave
E Colorado St
Wisconsin Ave
Molino Ave
E 3rd St
BLUFF HEIGHTS
Coronado Ave
Redondo Ave
E Vista St
4 MINUTES
[redacted] Ave, Long Beach, CA 90814, USA
5.0★ | POOL | 1.9X

Feature Extraction

- In surge prediction:
  - Location and time of past surges
  - Events
  - Number of people traveling to an area
  - Typical demand curves in an area
  - Demand in other areas

Data Cleaning

- Removing outliers
- Normalizing data
- Missing values
- …

Learning

- Build a predictor that best describes an outcome for the observed features

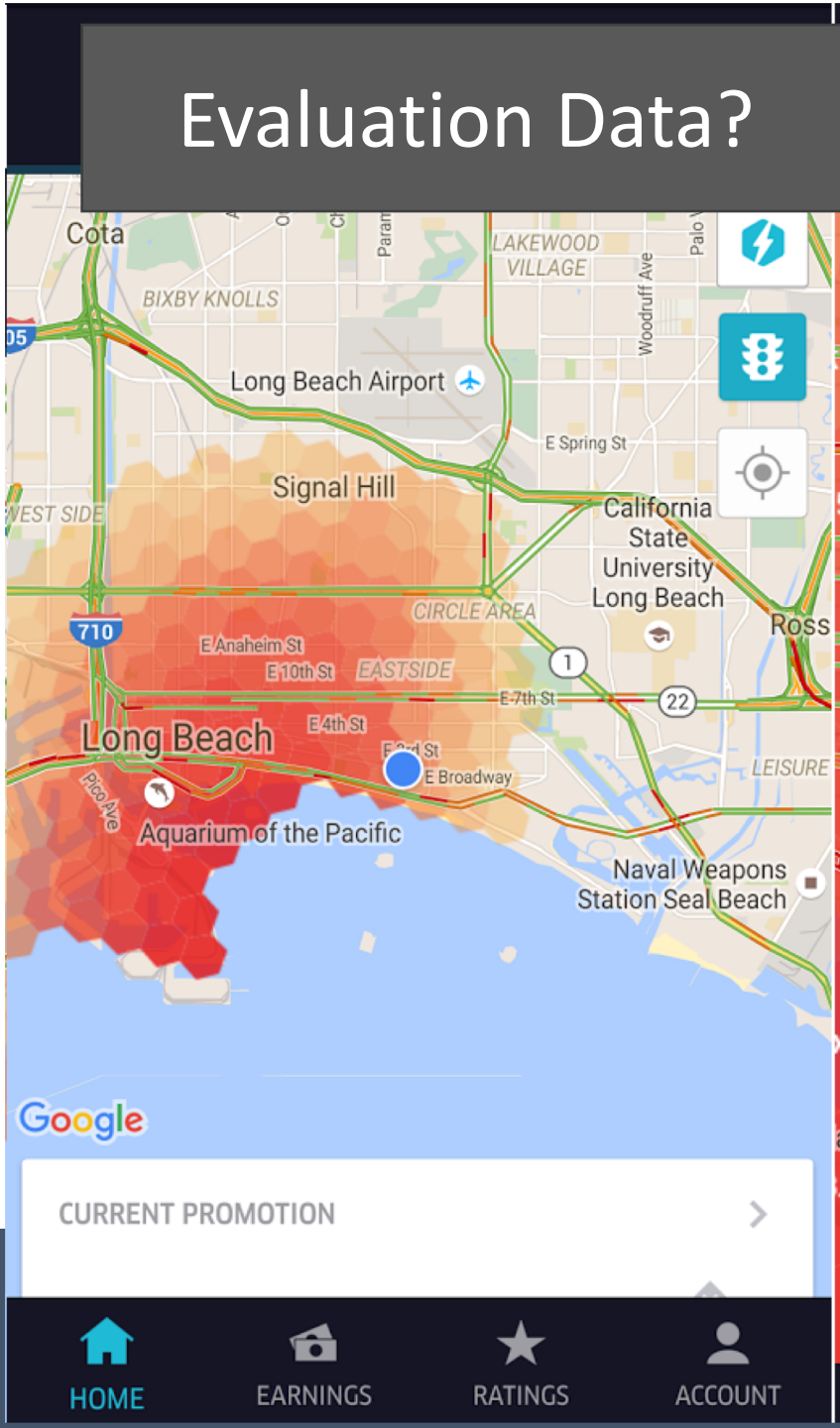
Evaluation

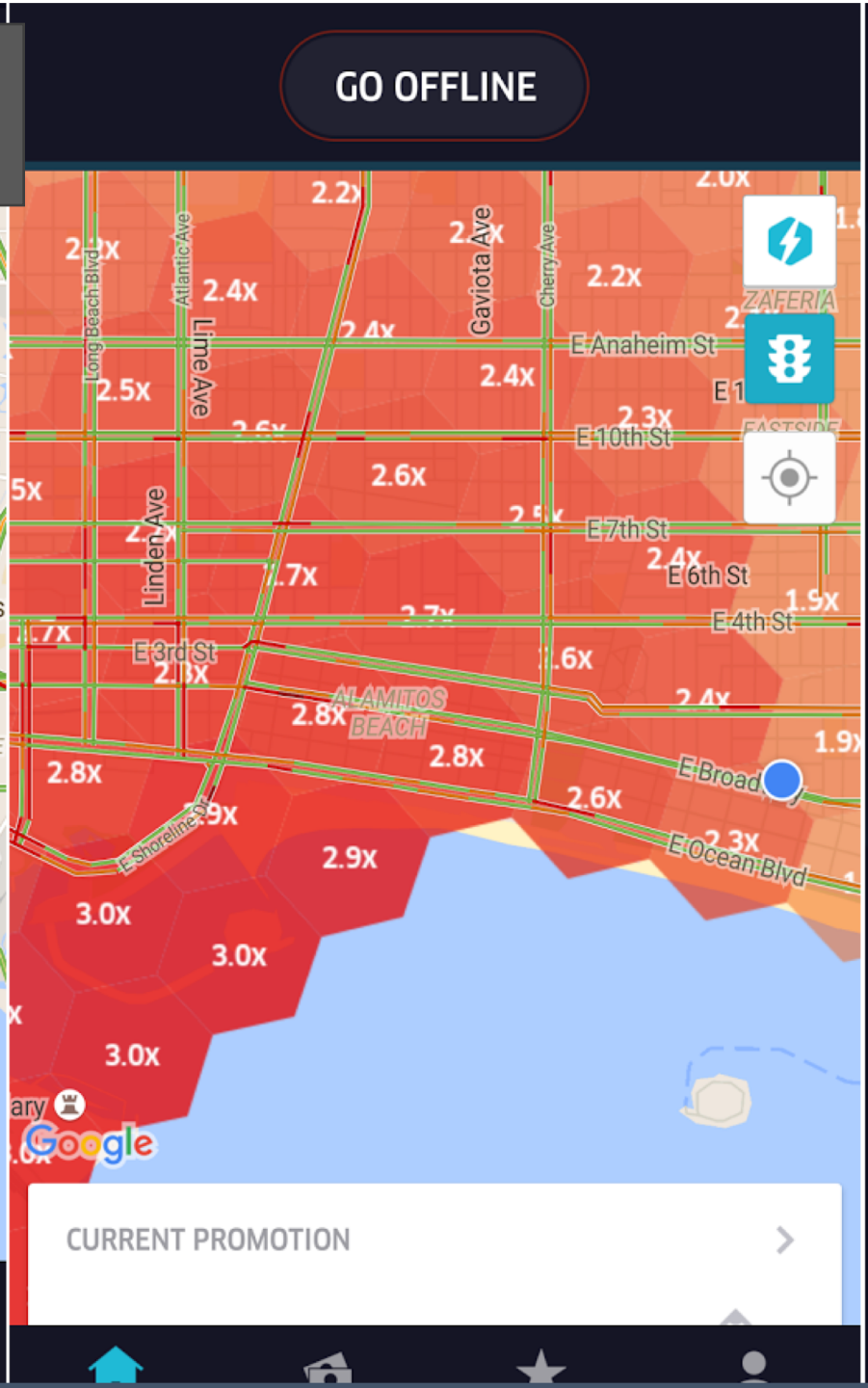
- Prediction accuracy on learned data vs
- Prediction accuracy on unseen data
  - Separate learning set, not used for training

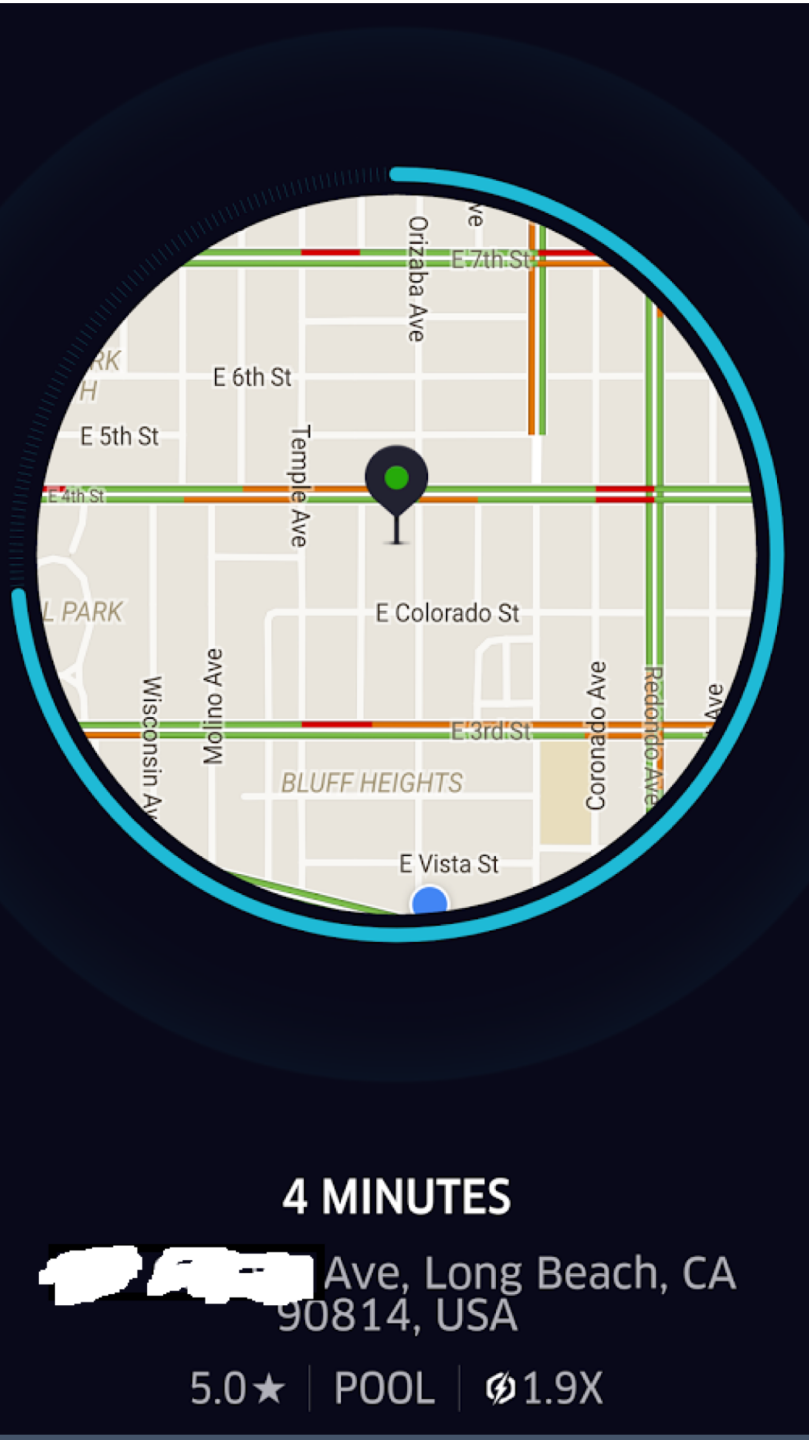
- For binary predictors: false positives vs false negatives, recall, precision
- For numeric predictors: average (relative) distance between real and predicted value
- For ranking predictors: topK etc

Evaluation Data?

Evaluation Data?







Learning and Evaluating in Production

- Beyond static data sets, **build telemetry**
- Design challenge: identify mistakes in practice

- Use sample of live data for evaluation
- Retrain models with sampled live data regularly
- Monitor performance and intervene

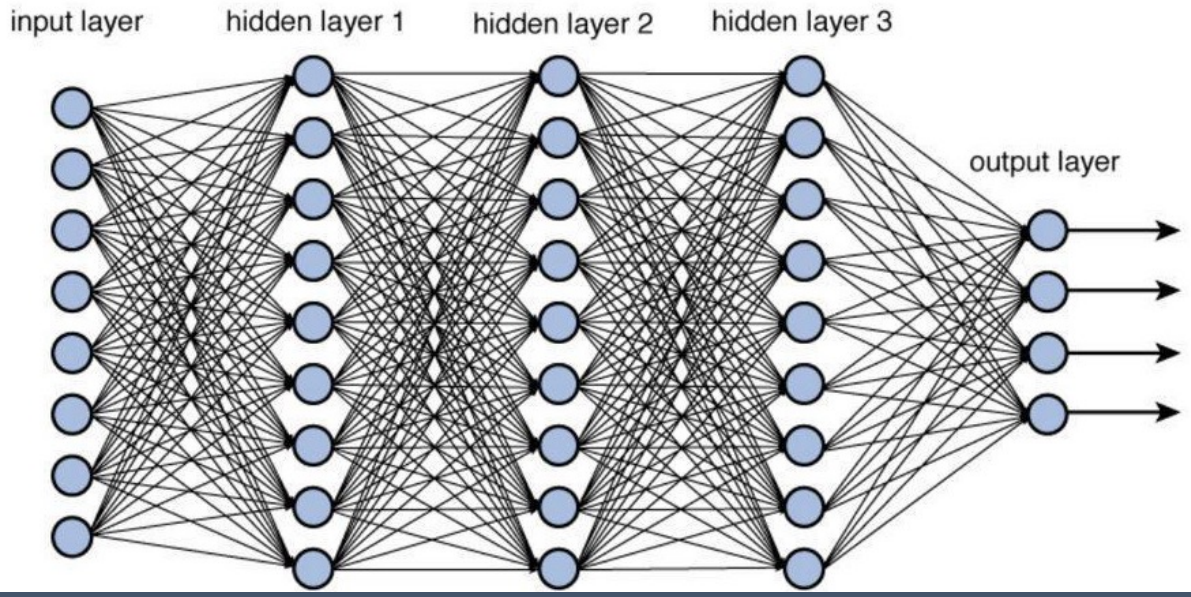
ML COMPONENT TRADEOFFS

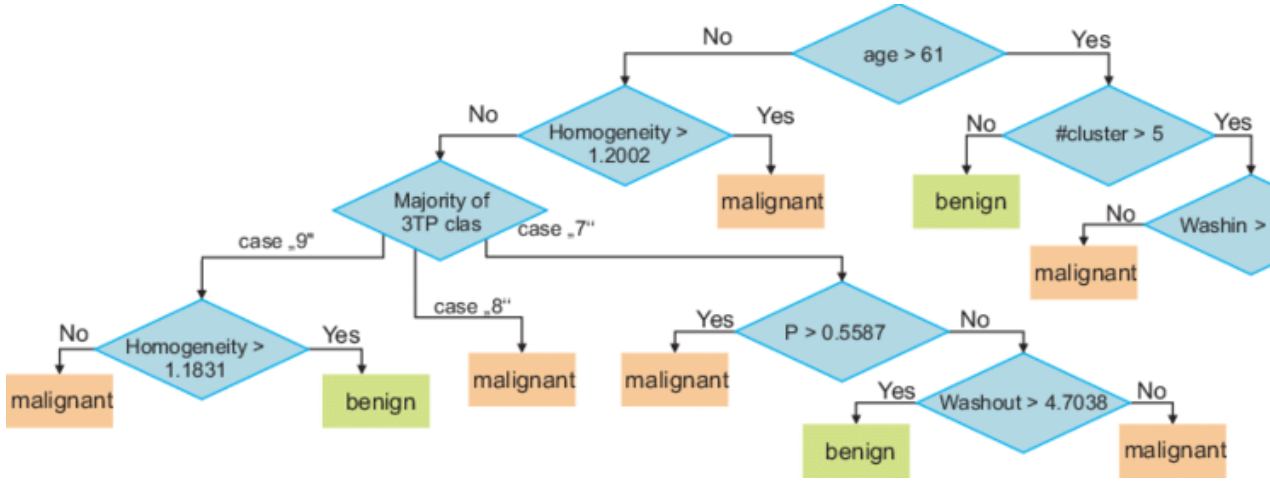
Qualities of ML Components

- Accuracy
- Capabilities (e.g. classification, recommendation, clustering…)
- Amount of training data needed
- Inference latency
- Learning latency; incremental learning?
- Model size
- Explainable? Robust?
- …

Understanding Capabilities and Tradeoffs

- Deep Neural Networks
- Decision Trees





SYSTEM ARCHITECTURE CONSIDERATIONS

Where should the model live?

Glasses

Phone

Cloud

OCR Component

Translation Component

Where should the model live?

Car

Phone

Cloud

Surge Prediction

Considerations

- How much data is needed as input for the model?
- How much output data is produced by the model?
- How fast/energy consuming is model execution?
- What latency is needed for the application?
- How big is the model? How often does it need to be updated?
- Cost of operating the model? (distribution + execution)
- Opportunities for telemetry?
- What happens if users are offline?

Typical Designs

- Static intelligence in the product
  - difficult to update
  - good execution latency
  - cheap operation
  - offline operation
  - no telemetry to evaluate and improve
- Client-side intelligence
  - updates costly/slow, out of sync problems
  - complexity in clients
  - offline operation, low execution latency

Typical Designs

- Server-centric intelligence
  - latency in model execution (remote calls)
  - easy to update and experiment
  - operation cost
  - no offline operation
- Back-end cached intelligence
  - precomputed common results
  - fast execution, partial offline
  - saves bandwidth, complicated updates
- Hybrid models

Other Considerations

- Coupling of ML pipeline parts
- Coupling with other parts of the system
- Ability for different developers and analysists to collaborate
- Support online experiments
- Ability to monitor

Reactive System Design Goals

- Responsive
  - consistent, high performance
- Resilient
  - maintain responsive in the face of failure, recovery, rollback
- Elastic
  - scale with varying loads

Common Design Strategies

- Message-driven, lazy computation, functional programming
  - asynchronous, message passing style
- Replication, containment, supervision
  - replicate and coordinate isolated components, e.g. with containers
- Data streams, “infinite data”, immutable facts
  - streaming technologies, data lakes

- See “big data systems” and “cloud computing”

Making Decisions

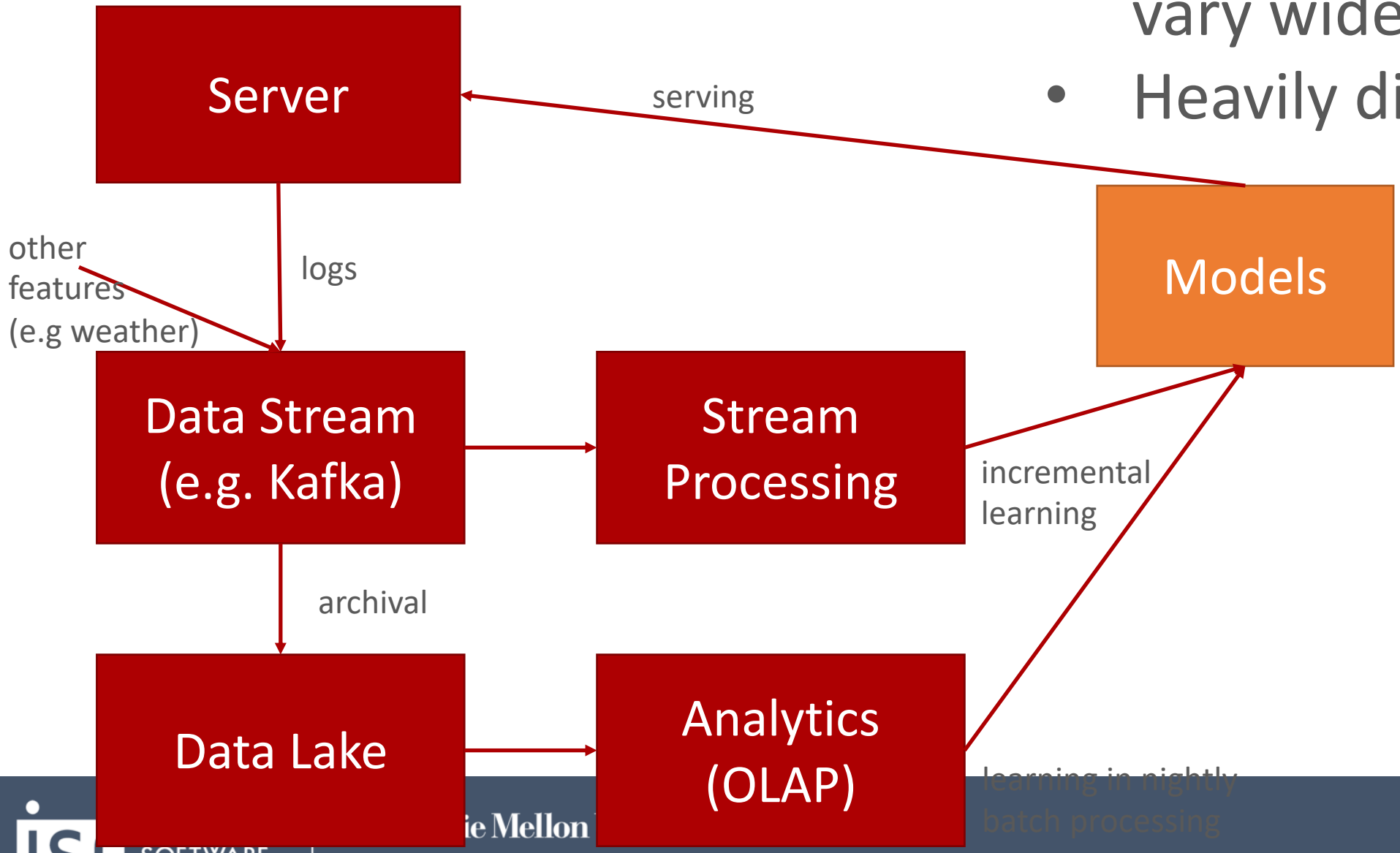
- What steps to take?
- What information to collect?

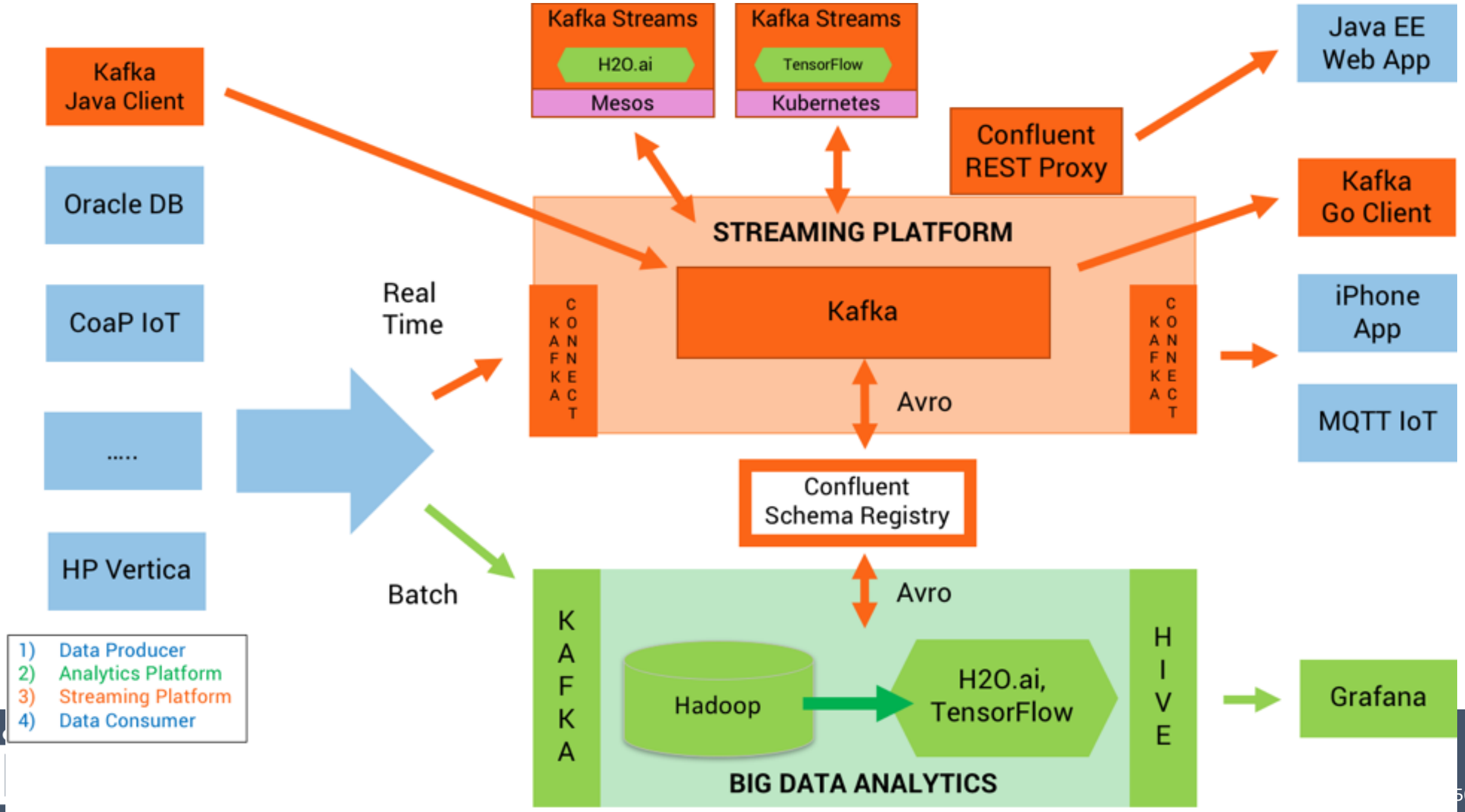
UPDATING MODELS

Updating Models

- Models are rarely static outside the lab
- Data drift, feedback loops, new features, new requirements
- When and how to update models?
- How to version? How to avoid mistakes?

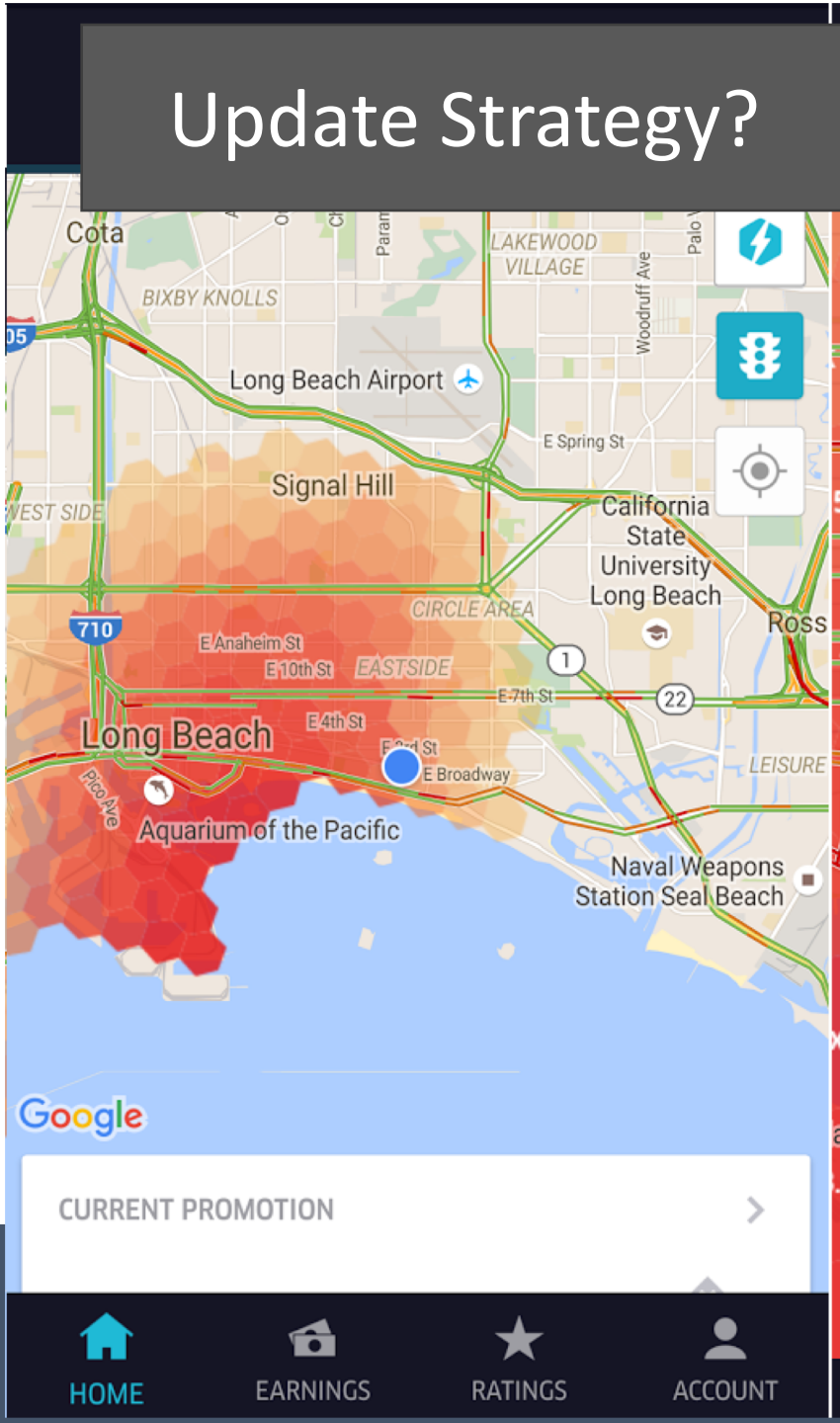
- Latency and automation vary widely
- Heavily distributed

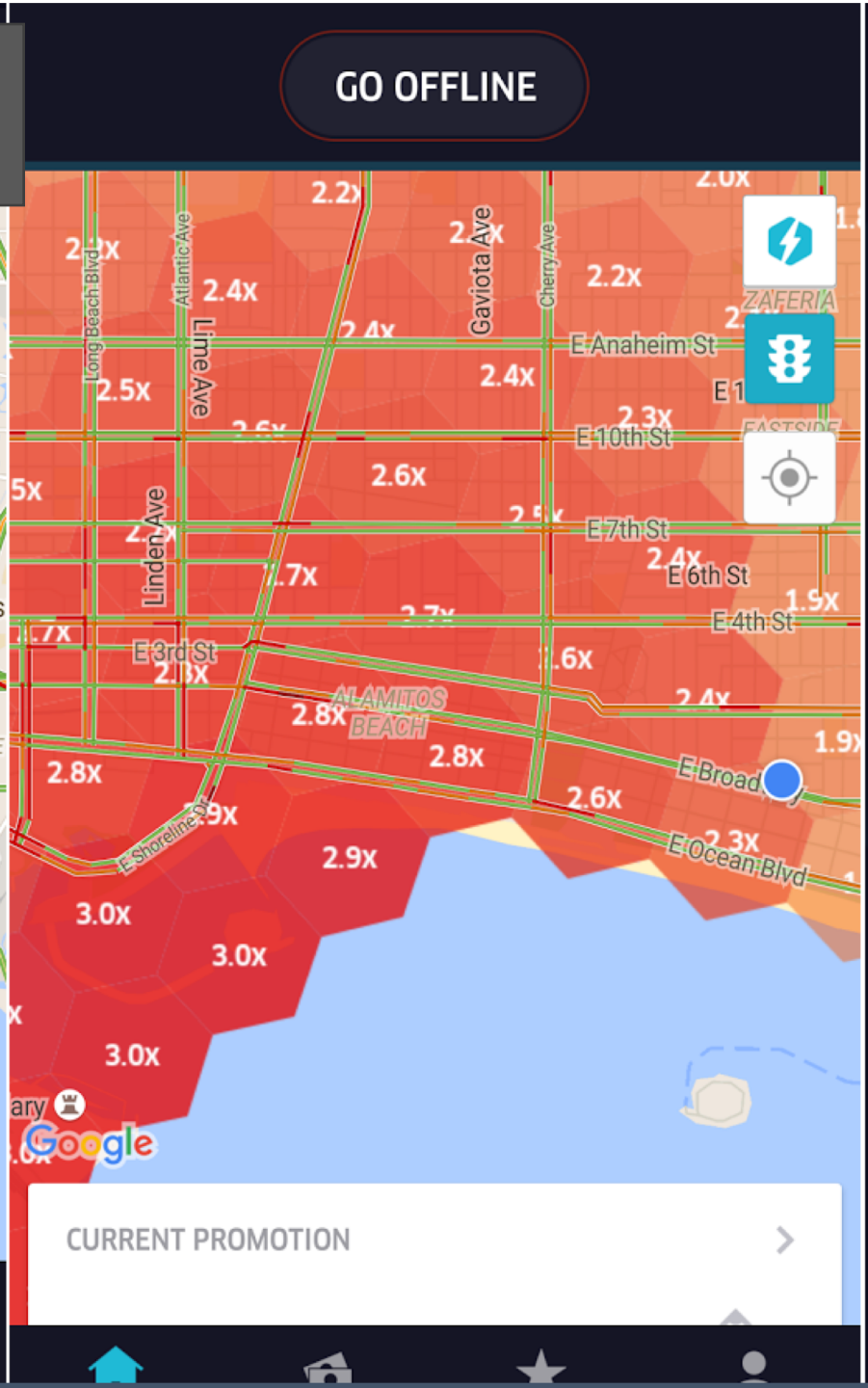


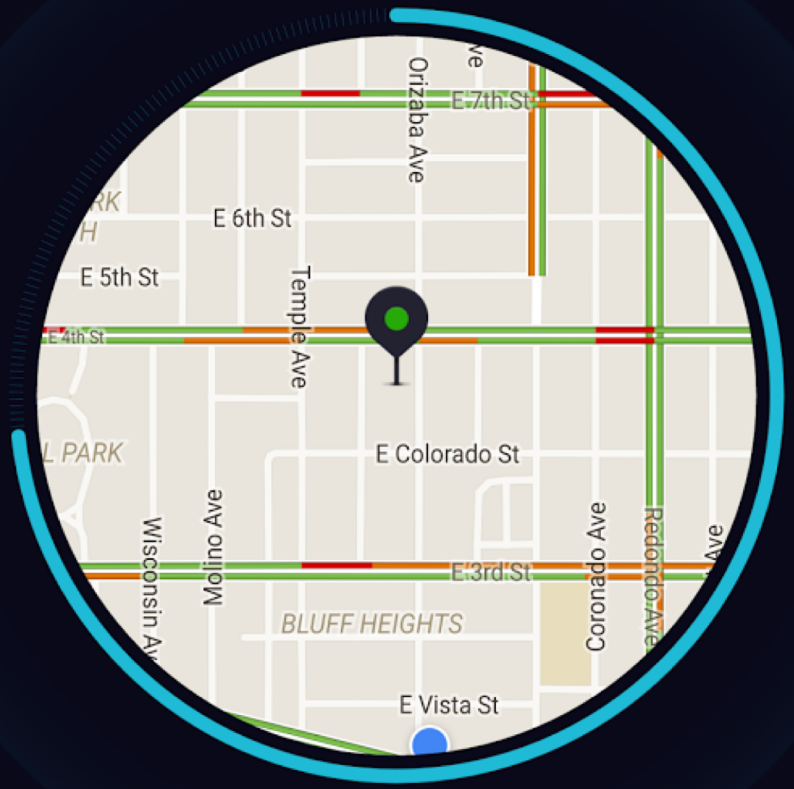

Kafka Java Client
Oracle DB
CoaP IoT
.....
HP Vertica
Real Time
Batch
Kafka Streams
H2O.ai
Mesos
Kafka Streams
TensorFlow
Kubernetes
Confluent REST Proxy
STREAMING PLATFORM
KAFKA CONNECT
Kafka
Avro
KAFKA CONNECT
Confluent Schema Registry
Avro
KAFKA
Hadoop
H2O.ai, TensorFlow
BIG DATA ANALYTICS
HIVE
Java EE Web App
Kafka Go Client
iPhone App
MQTT IoT
Grafana
1) Data Producer
2) Analytics Platform
3) Streaming Platform
4) Data Consumer

Update Strategy?

Update Strategy?







4 MINUTES

[redacted] Ave, Long Beach, CA 90814, USA

5.0★ | POOL | 1.9X

Making Decisions

- What steps to take?
- What information to collect?

PLANNING FOR MISTAKES

Mistakes will happen

- No specification
- ML components detect patterns from data (real and spurious)
- Predictions are often accurate, but mistakes always possible
- Mistakes are not predicable or explainable or similar to human mistakes
- Plan for mistakes
- Telemetry to learn about mistakes?

How Models can Break

- System outage
- Model outage
  - model tested? deployment and updates reliable? file corrupt?
- Model errors
- Model degradation
  - data drift, feedback loops

Hazard Analysis

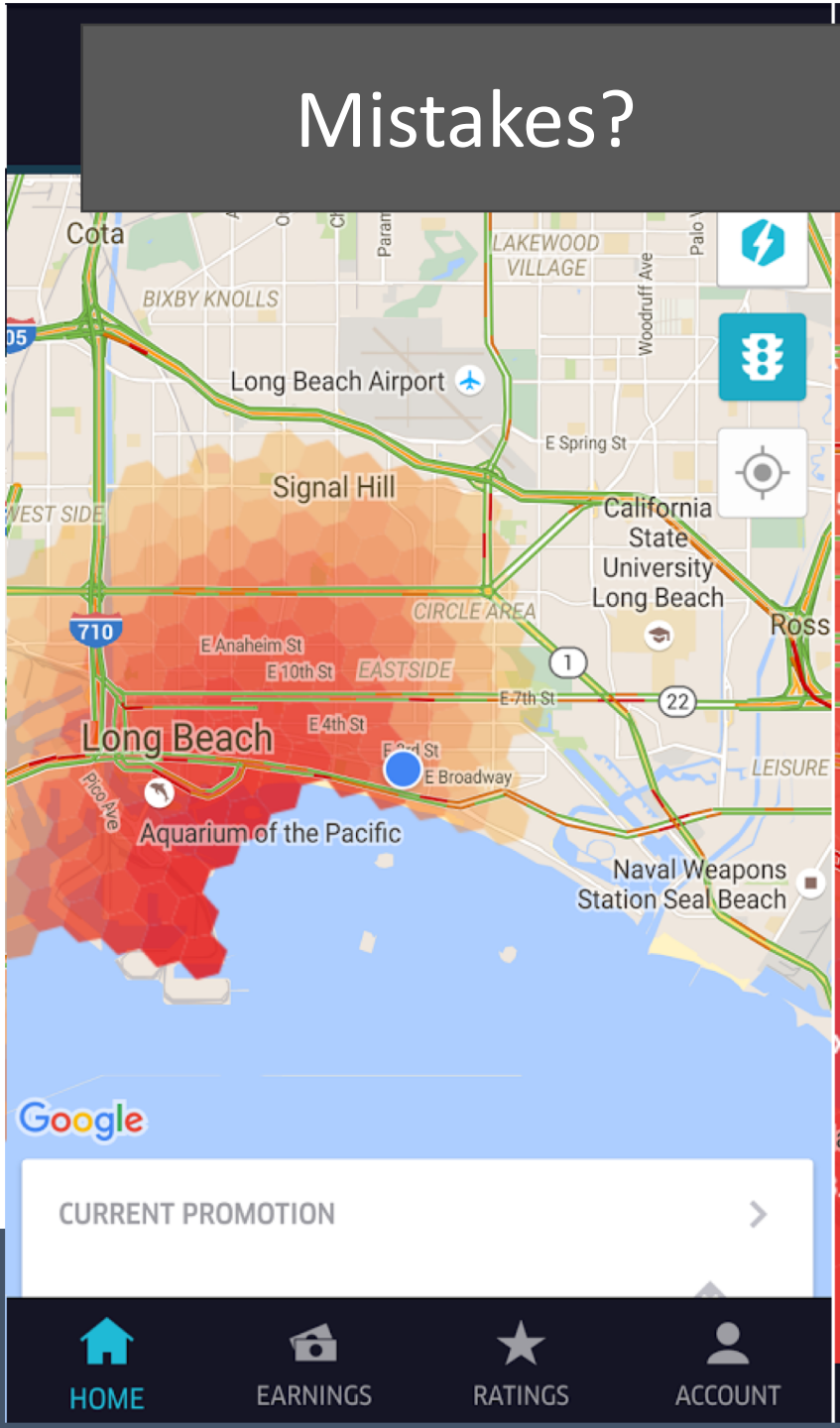
- Worst thing that can happen?
- Backup strategy? Undoable? Nontechnical compensation?

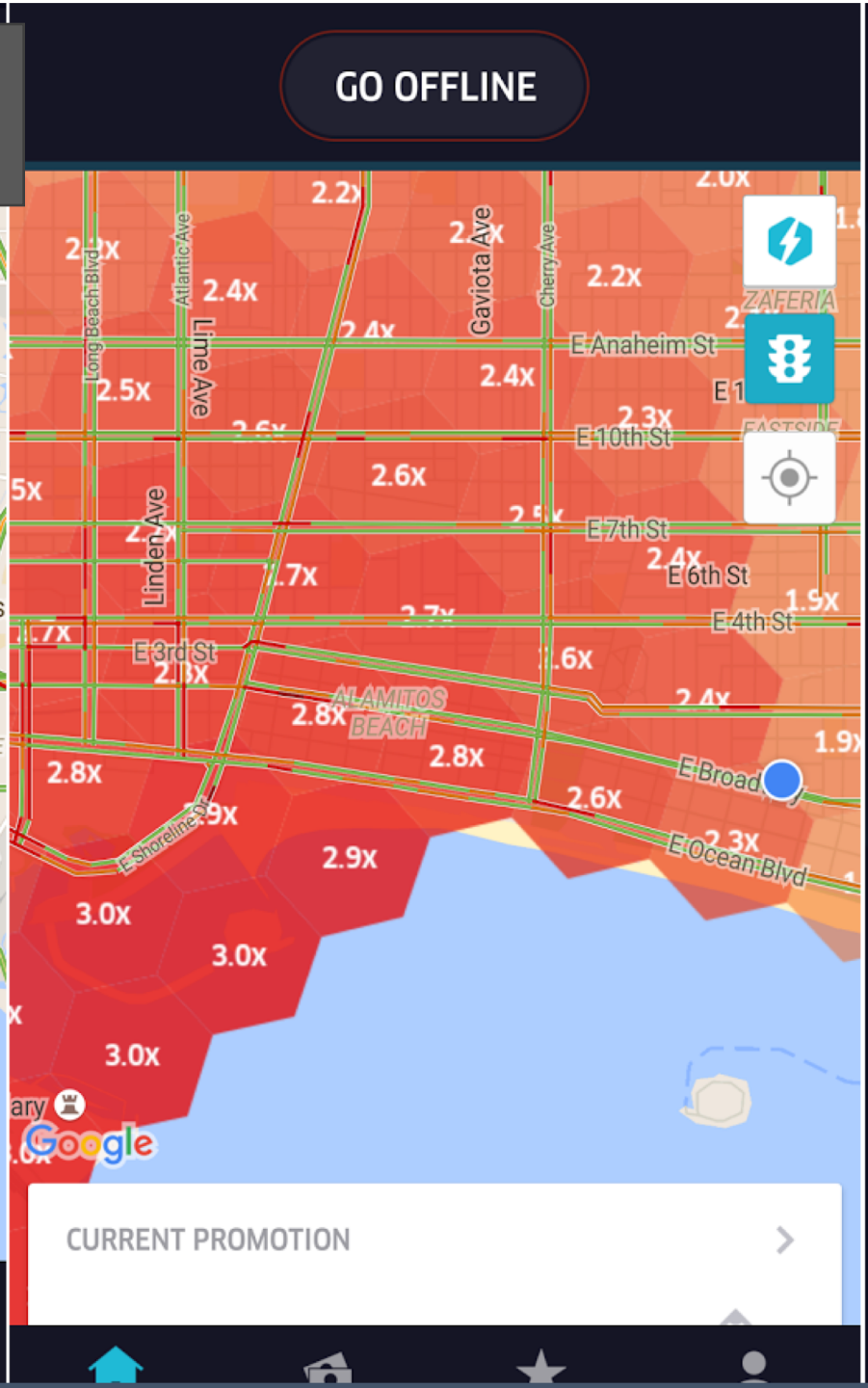
Mitigating Mistakes

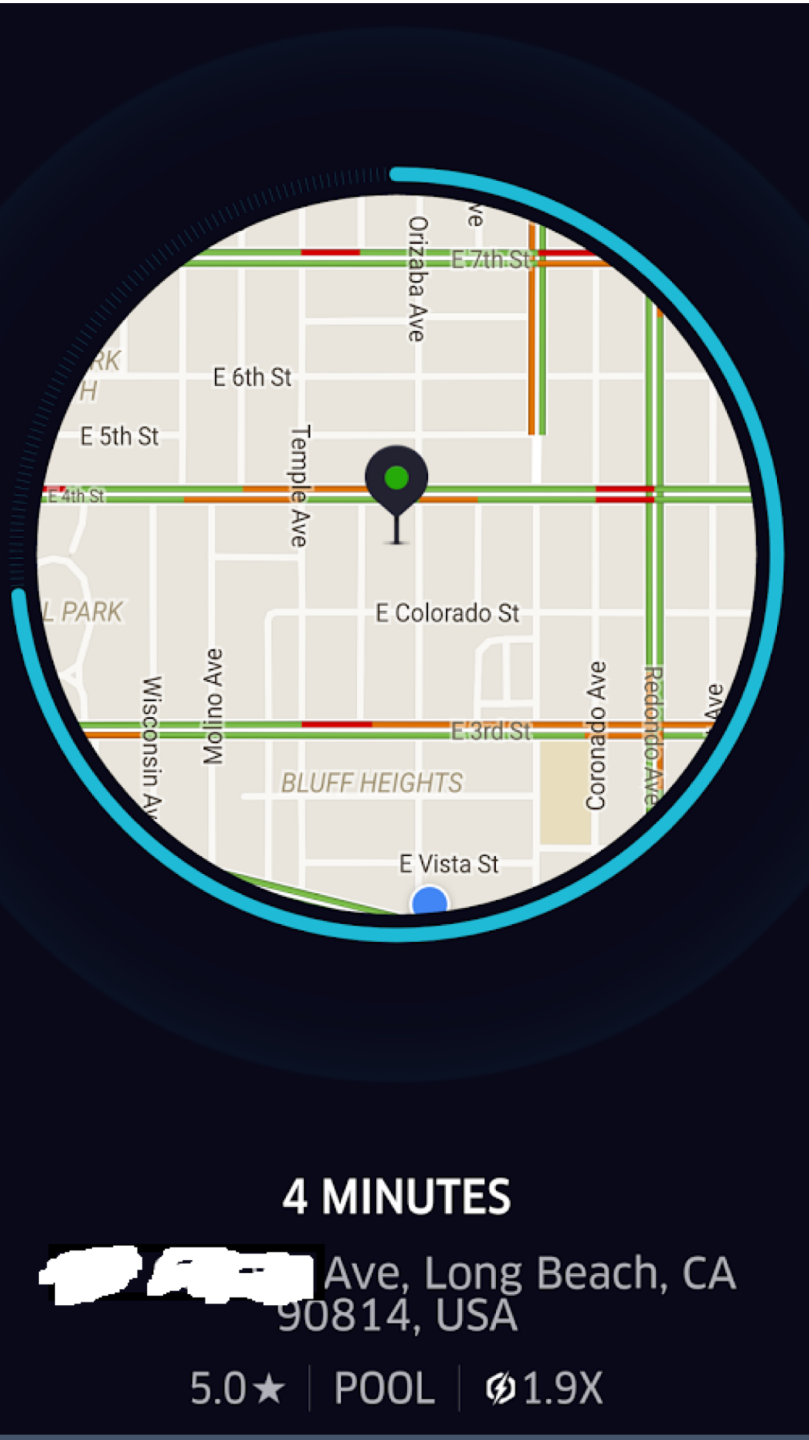
- Investigating in ML
  - e.g., more training data, better data, better features, better engineers
- Less forceful experience
  - e.g., prompt rather than automate decisions, turn off
- Adjust learning parameters
  - e.g., more frequent updates, manual adjustments
- Guardrails
  - e.g., heuristics and constraints on outputs
- Override errors
  - e.g., hardcode specific results

Mistakes?

Mistakes?


Cota
LAKEWOOD VILLAGE
BIXBY KNOLLS
Long Beach Airport
E Spring St
Signal Hill
California State University Long Beach
Ross
CIRCLE AREA
E Anaheim St
E 10th St
EASTSIDE
E 7th St
E 4th St
Long Beach
E Broadway
LEISURE
Aquarium of the Pacific
Naval Weapons Station Seal Beach
Google
CURRENT PROMOTION
HOME
EARNINGS
RATINGS
ACCOUNT


GO OFFLINE
Long Beach Blvd
Atlantic Ave
Lime Ave
Gaviota Ave
Cherry Ave
E Anaheim St
E 10th St
E 7th St
E 6th St
E 4th St
E 3rd St
Linden Ave
ALAMITOS BEACH
E Broadway
E Shoreline Dr
E Ocean Blvd
2.2x
2.4x
2.5x
2.6x
2.8x
2.9x
3.0x
Google
CURRENT PROMOTION


Orizaba Ave
E 7th St
E 6th St
E 5th St
Temple Ave
E Colorado St
Wisconsin Ave
Molino Ave
E 3rd St
BLUFF HEIGHTS
Coronado Ave
Redondo Ave
E Vista St
4 MINUTES
[redacted] Ave, Long Beach, CA 90814, USA
5.0★ | POOL | 1.9X

Telemetry

- Purpose:
  - monitor operation
  - monitor success (accuracy)
  - improve models over time (e.g., detect new features)
- Challenges:
  - too much data – sample, summarization, adjustable
  - hard to measure – intended outcome not observable? proxies?
  - rare events – important but hard to capture
  - cost – significant investment must show benefit
  - privacy – abstracting data

Model Orchestration

- Multiple models work together
  - ORC + statistical translation
  - driver activity predictor + demand predictor + event predictor + weather model
  - model + railguards
- Many different composition strategies
  - ensemble learning
  - integrated models
  - model sequencing
  - partitioning

Summary

- Machine learning in production systems is challenging
- Many tradeoffs in selecting ML components and in integrating them in larger system
- Plan for updates
- Manage mistakes, plan for telemetry