1. ML in Uber Eats
   - Goals & Challenges
   - ML Platform @ Uber

2. How Time Predictions Power Dispatch System

3. Deep Dive in Time Predictions
   - Food Preparation Time Prediction
   - Delivery Time Estimation
   - Travel Time Estimation

4. Q&A
ML in Uber Eats
Agenda

- Goals & Challenges
- ML Platform @ Uber
Our Scale

> 500 Cities

> 220,000 Restaurant Partners

~ 8B Gross Bookings for 2018
Make eating well **effortless**, every day, for **everyone**.
Goals & Challenges

Reliable
Predicting the Future

Affordable
Network Efficiency

Effortless
Food Discovery
Eyeball ETD prediction

ETD prediction

Prep-time prediction

ETA prediction

eater browsing

order created

dispatch

delivery-partner arrival

food ready

delivery-partner begins trip

delivery-partner arrival

food dropped-off
ML Platform @ Uber
Feature Report
Model Accuracy Report

Test Data Performance

- Threshold: 0.5884
- ROC: 0.268
- AUC: 0.7936
- Error: 0.4907

Confusion Matrix

<table>
<thead>
<tr>
<th></th>
<th>True Positive</th>
<th>False Negative</th>
<th>True Negative</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>0.21</td>
<td>0.093</td>
<td>0.86</td>
<td>0.18</td>
</tr>
</tbody>
</table>

The reliability diagram shows how reliable (or "well-calibrated") the model's probability estimates are when evaluated on the test data. For example, a well-calibrated (binary) model should classify the samples such that among the samples to which it gives a probability close to 0.8 of belonging to the positive class, approximately 80% of those samples actually belong to the positive class.

More Info
How Time Predictions Power Dispatch System
Agenda

- Overview of Dispatch System
- Evolution via Time Predictions
  - Dispatch System w/o Time Predictions
  - Dispatch System w/ Time Predictions
Make Demand-Supply Matching Decisions

Challenges

- Solve an NP-Hard problem with a large problem space within seconds
- Improve efficiency without compromising delivery quality
- Eater & Restaurant & Delivery Partner
Eater & Restaurant & Delivery Partner

Eater
- Fast drop-off
- Low delivery fee
- 24/7

Restaurant Partner
- Short wait time
- Low Unfulfillment

Delivery Partner
- Short wait time
- Smart route planning
- Quick hand-off
Matching Algorithm: An Augmented Vehicle Routing Problem (VRP)

\[
\text{Input}(\text{Plans}(\text{Supplies, Jobs, Constraints})) \implies \max \sum_{p \in \text{plans}} \text{DOF}(p) \implies \text{optimal plans}
\]

**DOF**: dispatch objective function

**Supply**: A courier eligible for job assignments

**Job**: A ordered list of waypoints (pickup, dropoff)

**Plan**: a combination of a supply and job(s)
Dispatch System w/o Time Predictions
When to Dispatch?

Order Created: 8:30

1st dispatch attempt: 8:53

Scheduled pick-up time: 9:00

Fixed 7 mins

do not dispatch a delivery-partner

dispatch a delivery-partner
How to Dispatch? (Greedy)

- Jobs dispatched independently without considering other jobs.
Before...

- Where is my food?
- Food is cold
- How much longer do I have to wait?
Dispatch System w/ Time Predictions
When to Dispatch?

Order created  
8:30  ...  8:50  ...  8:56  9:00

1st dispatch attempt

nth dispatch attempt

Predicted pick-up time

do not dispatch a driver

dispatch a driver
How to Dispatch? (Global)

- All jobs and supplies are considered at the same time.
Then we solve the entire set of jobs and supplies as a single global optimization problem.
**Greedy**

1 MIN +

5 MIN

6 MIN

---

**Global**

2 MIN +

2 MIN

4 MIN
After...

- Fast delivery times
- Accurate ETD estimations
- Track food location

- Reduce waiting at restaurants
- Maximize earning potential
- Be aware of estimated travel time

- Prevent delivery partners from waiting around
- Prevent food waiting for delivery partners
- Track delivery partner’s location

- Dispatch delivery partners at the right time
- Maintain supply/demand, prevent surge
Deep Dive in Time Predictions
Agenda

- Food Preparation Time Prediction
- Delivery Time Estimation
- Travel Time Estimation
Food Preparation Time Prediction
Why is Predicting Food Prep-time Difficult?

- 1) True restaurant prep-time is unknown!
  - Example: We need to infer true prep-time in a retrospective manner based on restaurants and delivery partners’ signals.

- 2) Prediction with limited signals
  - Example: The busyness in the actual restaurant is unknown
How Did We Use ML to Solve the Problem?

- Feature engineering
- ML Model
- Feedback Loop
Feature Engineering

- Historical features
  - Avg prep-time for 1 week, ...

- Real-time (Contextual) features
  - Time of day, day of week, order size, location, ...

- Near real-time features
  - Avg prep-time for last 10 mins, ...
Representation Learning
SensorSignals
Conditional Random Field Model
Feature Engineering (Cont’d) - Data Pipeline

Data preparation pipelines push data into the Feature Store tables and training data repositories.

- Time of day, day of week, order size, location, ...
- Preparation time
- Sensing & Perception
- Bluetooth Data
ML Model

- Model: Gradient boosting decision trees (XGBoost)
- Historical features
- Realtime (Contextual) features
- Near real-time features
Hyperparameter tuning

Image source: [www.nature.com/articles/nature14541](http://www.nature.com/articles/nature14541)
Model training jobs use Feature Store and training data repository data sets to train models and then push them to the model repository.
Model Training (Cont’d) - Model Deployment
Model Training (Cont’d) - Make Predictions
**ML Model with Feedback Loop**

**Historical features**
E.g. average prep-time in last week

**Near real time features**
E.g. average prep-time in last few minutes

**Real time features**
E.g. order size, time of day

**Production model (GBDT)**

**Michelangelo model training**

**Predicted prep-time**

**Updated Data**

---

**Online prediction**

**Offline training**
Future Improvements

- Ground truth exploration
  - Experiment in restaurants
  - ...
- Improving ML model
  - Feature engineering
    - Exploration of places, weather, and event data
    - Model partitioning
    - ...
  - Leverage ensemble learning (stacking)
  - Collaboration with AI Labs on more deep learning models
Delivery Time Estimation
Eater-facing ETD

- EATER
  - 0) eyeball
  - 1) order created
  - 10) food received

- RESTAURANT
  - 2) order accepted
  - 6) food ready

- DELIVERY PARTNER
  - 3) dispatched
  - 4) accepted trip
  - 5) arrived
  - 7) departed
  - 8) begun trip
  - 9) arrived
  - 11) ended trip

- observable state
- not-observable state
Why is predicting ETD difficult?

0) eyeball 
1) order created 
2) order accepted 
3) dispatched 
4) accepted trip 
5) arrived 
6) food ready
7) departed 
8) begun trip 
9) arrived 
10) food received 
11) ended trip
Travel Time Estimation
Credits
Teams @ Uber

Special thanks to:

- Engineers
- Data Scientists
- Product managers
- Product Ops
- Data Analysts
THANK YOU

Q & A
Proprietary and confidential © 2019 Uber Technologies, Inc. All rights reserved. No part of this document may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval systems, without permission in writing from Uber. This document is intended only for the use of the individual or entity to whom it is addressed and contains information that is privileged, confidential or otherwise exempt from disclosure under applicable law. All recipients of this document are notified that the information contained herein includes proprietary and confidential information of Uber, and recipient may not use, disseminate, or in any way disclose this document or any of the enclosed information to any person other than employees of addressee to the extent necessary for consultations with authorized personnel of Uber.