Food Discovery with Uber Eats:

Holistic Multi-Objective Optimization for the Marketplace

Uber | **Yuyan Wang, 06/28/2019**

About me

Currently, I am:

- Senior data scientist at Uber Eats
    - Personalized ranking and recommendation algorithms
        - Multi-objective optimization for the three-sided marketplace
        - Holistic ranking with heterogeneous & hierarchical contents

About me

Currently, I am:

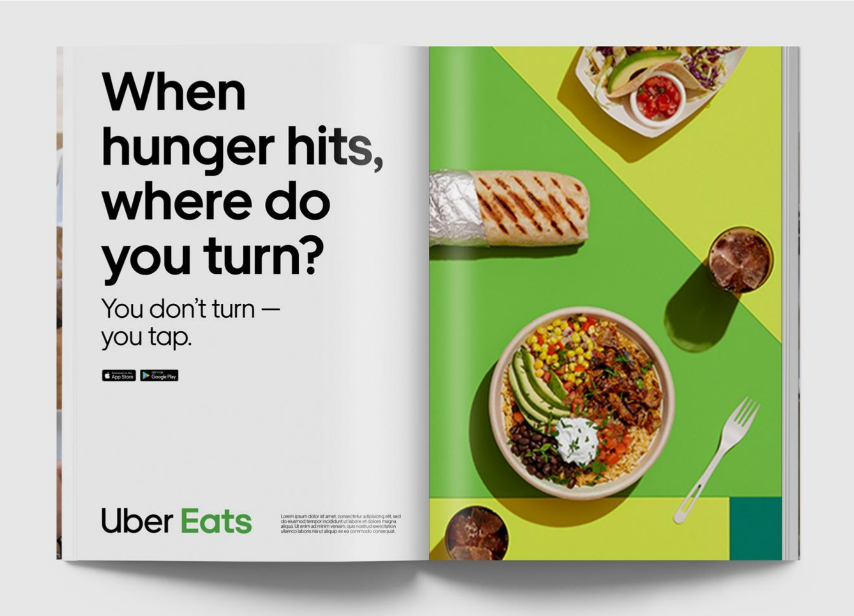
- Senior data scientist at Uber Eats
  - Personalized ranking and recommendation algorithms
    - Multi-objective optimization for the three-sided marketplace
    - Holistic ranking with heterogeneous & hierarchical contents

Before Uber, I was:

- Ph.D in statistics from Princeton University
  - Thesis: Robust high-dimensional regression and factor models
- BS.c in statistics from USTC

Outline

- Problem space: Ranking & recommendation
- Multi-Objective Optimization (MOO)
- Holistic optimization: Heterogeneous and hierarchical Content
- Outside ranking: Experimentation



Uber Eats 优食

Uber Eats 優食

أوبر إيتس
Uber Eats

Ranking and recommendation: We are unique

3-sided marketplace

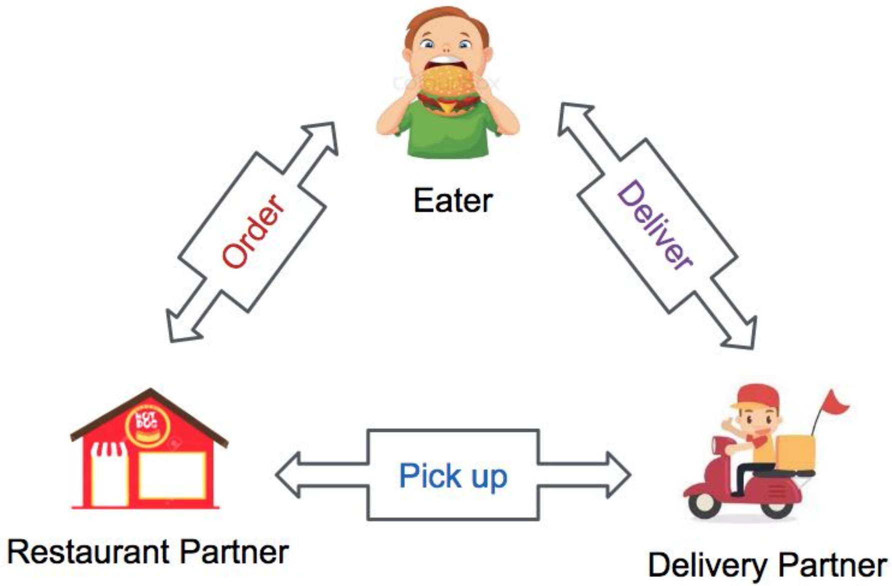


Image Source: https://www.shutterstock.com

Ranking and recommendation: We are unique

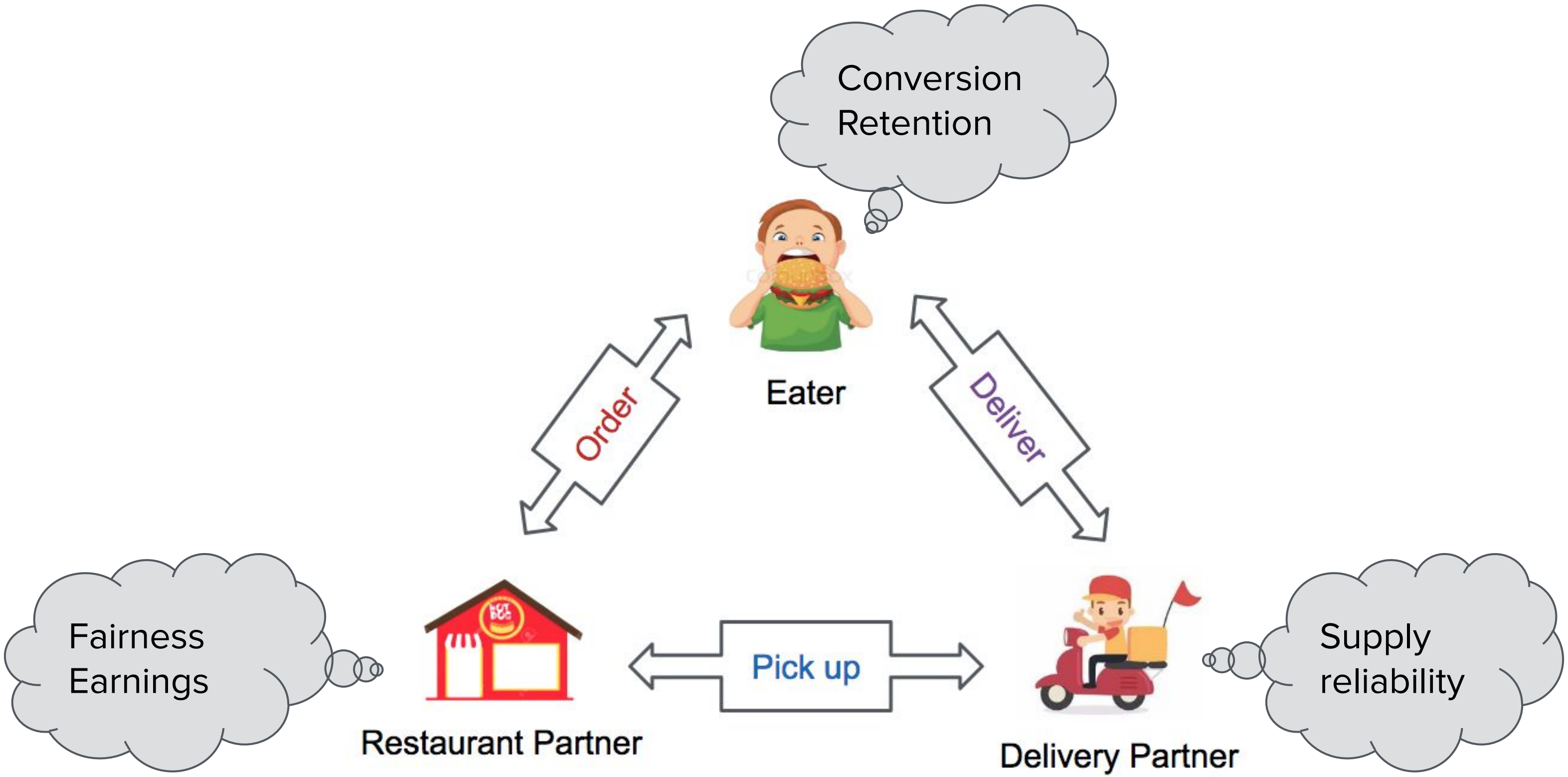
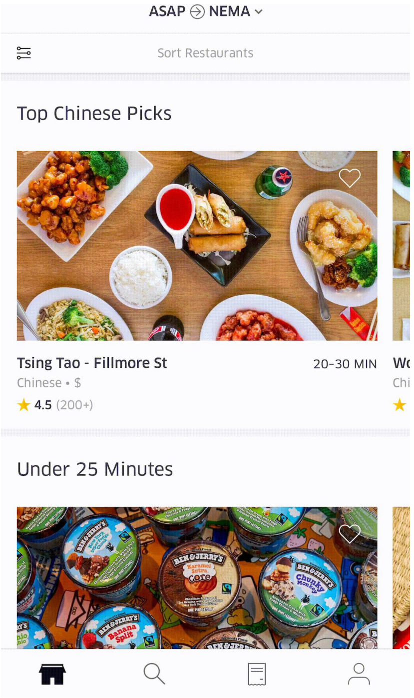
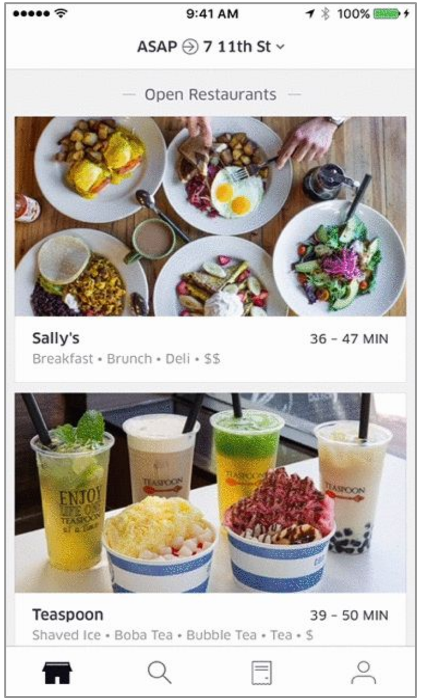


Image Source: https://www.shutterstock.com

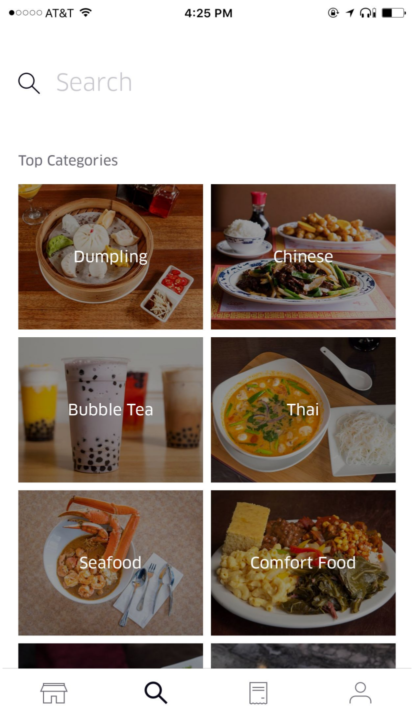
Another challenge: Heterogeneous & hierarchical content



Row Ranking



Vertical Ranking

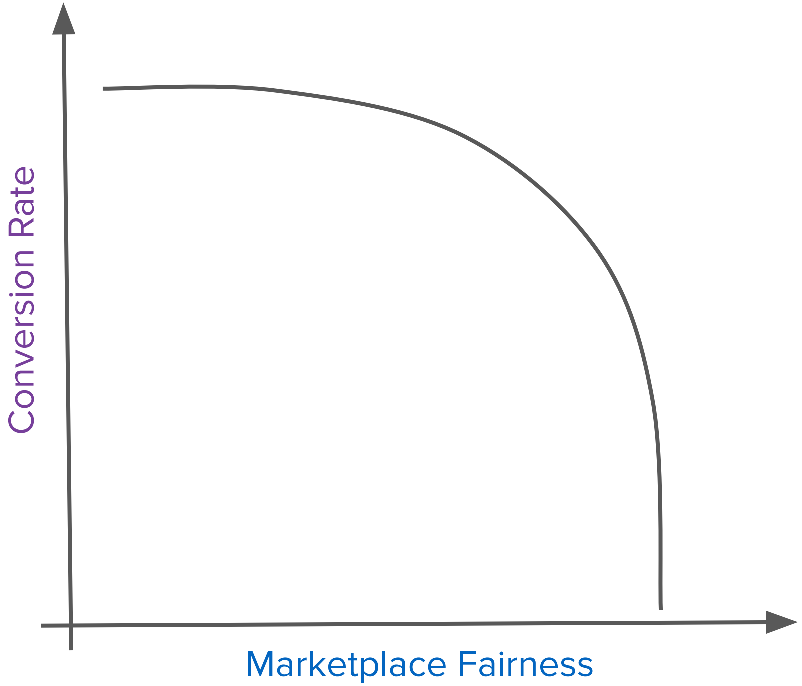


Search

MOO: Ranking to Serve the Marketplace

- Conventional ML model
  - Single objective: eater
  - GBDT, XGBoost

- **Multi-objective Optimization**:
  - Multiple objectives:
    eater / restaurant / delivery partner
  - Tradeoff
  - Linear / Quadratic programming



MOO: Building a Fair Marketplace

vs.

Well-established Restaurants

New/Low-volume Restaurants

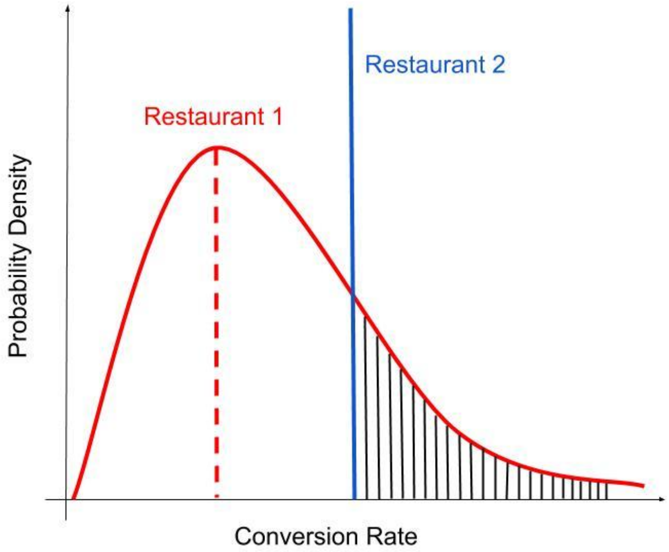
Image Source: https://thehalalguys.com/, http://www.istockphoto.com/

Explore-Exploit with Multi-Armed Bandit

- Explore/exploit **tradeoff**
  - Exploit: restaurants with high predicted conv
  - Explore: new / low-volume restaurants

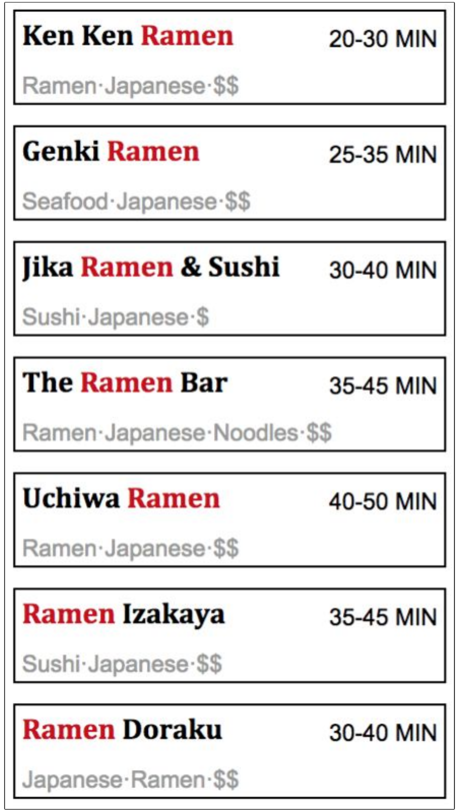
Explore-Exploit with Multi-Armed Bandit

- Explore/exploit **tradeoff**
  - Exploit: restaurants with high predicted conv
  - Explore: new / low-volume restaurants
- Bayesian modeling for posterior variance
  - New / low volume restaurant - High var
  - Well-established restaurant - Low var
  - Beta-binomial distribution
- Multi-armed bandit
  - ML model to estimate the mode of conversion
  - Bandit algorithm for explore-exploit (UCB/Thompson sampling)



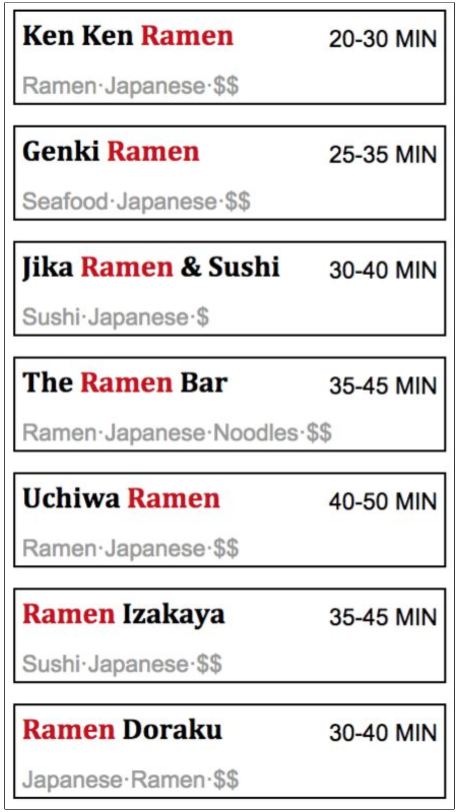
Relevance vs. Diversity

- What's wrong with ranking wrt relevance?
  - Overwhelming restaurants that are too similar to each other in a consecutive order
  - Recommendations should be both **accurate** and **diverse**



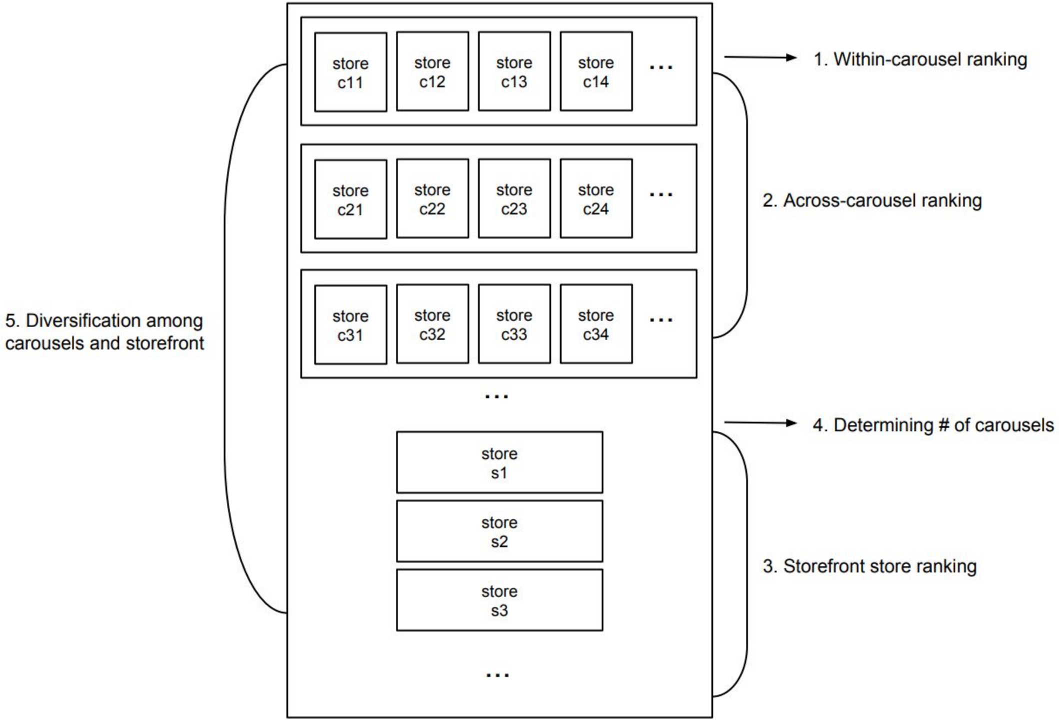
Relevance vs. Diversity

- What's wrong with ranking wrt relevance?
    - Overwhelming restaurants that are too similar to each other in a consecutive order
    - Recommendations should be both **accurate** and **diverse**
- Personalized diversification algorithm
    - Eater representation (taste profile)
    - Restaurant representation (cuisine profile)
    - Sequential/greedy optimization



With MOO in place, what’s next?

HRank: Holistic optimization of heterogeneous and hierarchical contents



HRank: Methodology

- A personalized, holistic approach for optimal home feed layout

- Key model components / assumptions
  - Triplet model: ***(eater, store, source)***
  - Scrolling discounting factor
    - User has limited patience
    - Within-row
    - Across-row

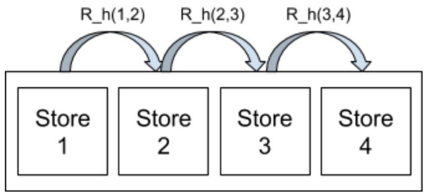


Figure 4: Horizontal impression discounting factor.

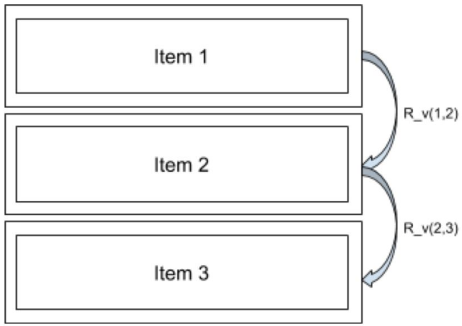


Figure 5: Vertical impression discounting factor.

HRank: Methodology

- A personalized, holistic approach for optimal home feed layout

- Key model components / assumptions
  - Triplet model: ***(eater, store, source)***
  - Scrolling discounting factor
    - User has limited patience
    - Within-row
    - Across-row
  - **Law of total probability**:
    Holistically optimize for **session-level** conversion
    (as opposed to **impression-level** conversion)

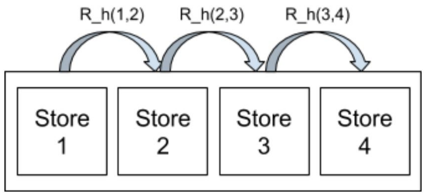


Figure 4: Horizontal impression discounting factor.

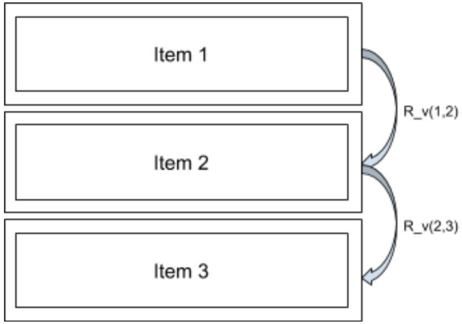


Figure 5: Vertical impression discounting factor.

$$P(O_{ic}=1) = p_1 + (1-p_1)\underline{R_h(1,2)}p_2 + (1-p_1)(1-p_2)\underline{R_h(1,3)}p_3 + \ldots + (1-p_1)\cdots(1-p_{N-1})\underline{R_h(1,N)}p_N .$$

Ongoing efforts: Holistic Multi-Objective Optimization

- Challenge 1: Heterogeneous & Hierarchical Contents

  => Need a holistic optimization and recommendation framework

- Challenge 2: Three-sided marketplace

  => Trade-off among different objectives

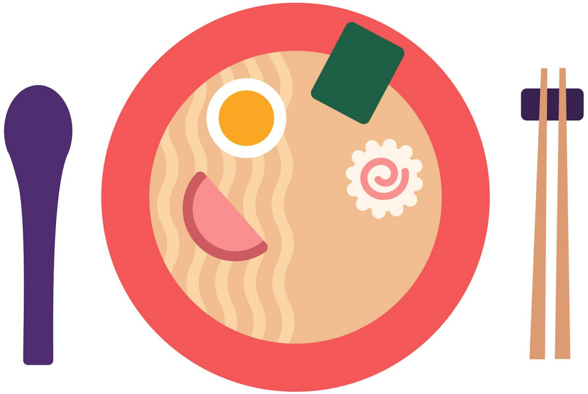
HRank

MOO

HMOO

Summary

- Problem space: ranking and recommendation
- Multi-Objective Optimization (MOO)
    - Building a Fair Marketplace
    - Relevance vs. Diversity
- Holistic Optimization
- Outside ranking: experimentation efforts



Experimentation efforts

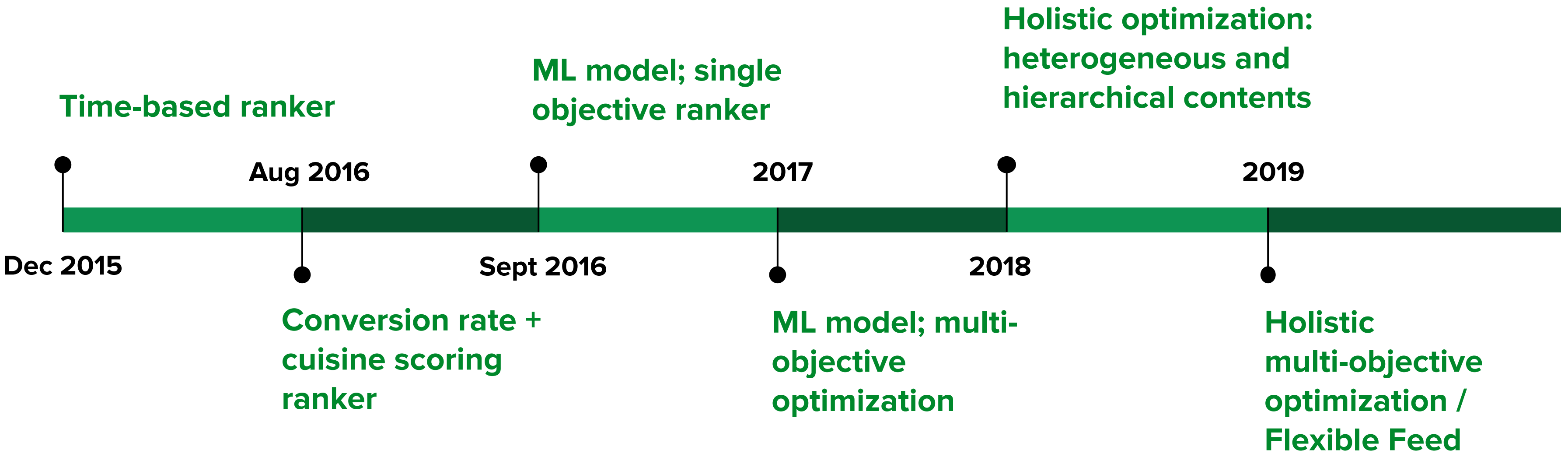
- Developed experimentation Python Notebook
  - 30+ metrics including topline business metrics and engagement metrics
  - A/B testing, A/A testing, segmentation analysis and coupled experiment analysis
  - Used across whole Uber Eats team by data scientists, engineers and product analysts
- Pushing A/B testing at Uber Eats to next level
  - Authored A/B testing guidelines and Uber Eats Data Science experimentation playbook
  - 10+ tutorials/talks on experimentation best practices

Thank you.

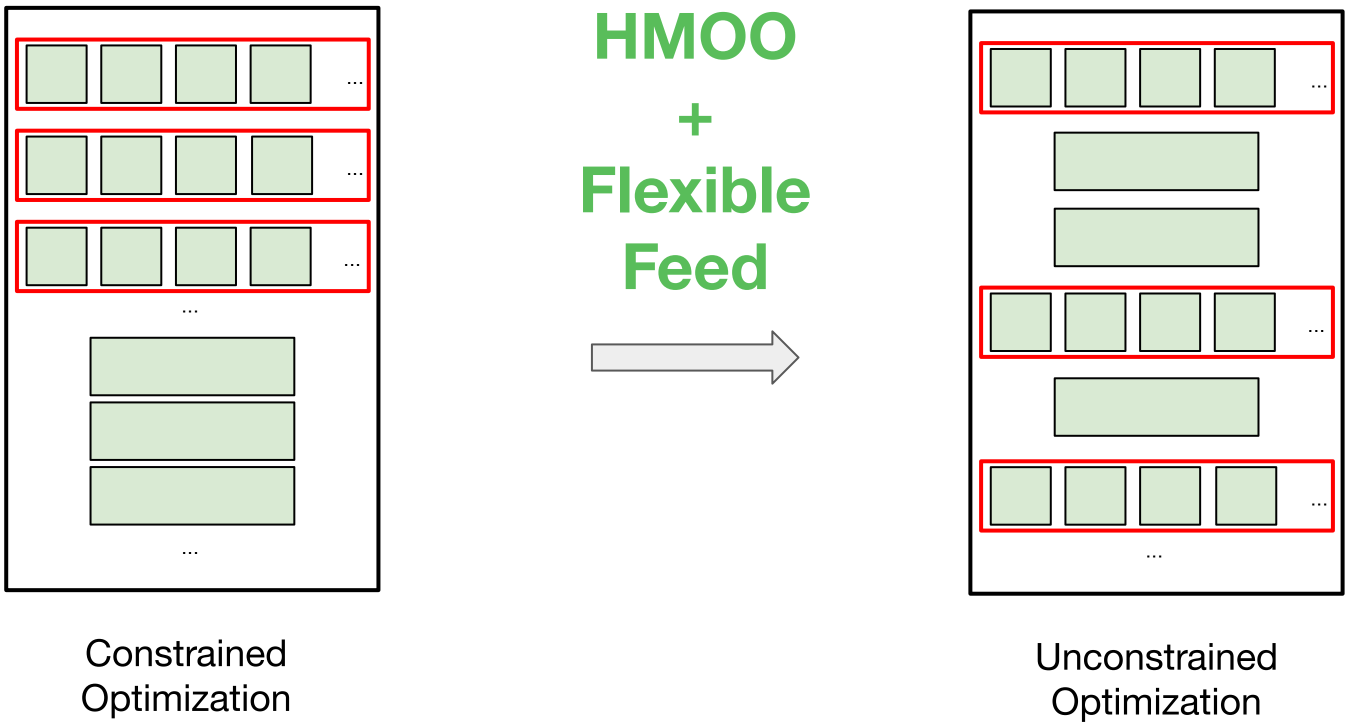
Uber Eats

Appendix

The Journey of Ranking



Flexible Feed: Unlocking more potential for HMOO



MOO: Multi-Objective Optimization

- Formulation
  - Expected total # of orders:

$$T(X) = \sum_i \sum_j x_{ij} p_{ij}$$

  - Expected total gross booking fee:

$$G(X) = \sum_i \sum_j x_{ij} p_{ij} g_j$$

  - Linear programming (LP):

$$\begin{gathered} max_X \; G(X) \\ s.t. \; T(X) \geq \alpha \cdot T(X^*) \\ x_{ij} \geq 0, \forall \, i, j \\ \sum_j x_{ij} = 1, \forall \, i \end{gathered}$$

Notation	Meaning
i	Indexes users
j	Indexes stores (restaurants)
$X = \{x_{ij}\}$	Serving plan (variable to be optimized)
x_{ij}	Probability that the serving scheme will recommend j th store to user i
$Q = \{q_{ij}\}$	Uniform serving plan with $q_{ij} = \frac{1}{J}$, where J is the total number of open stores.
p_{ij}	Estimated **conversion rate** of user i on store j
g_j	Estimated **gross booking fee** for store j (can be personalized as g_{ij})

A **huge** linear programming problem
-> scalability issue!

MOO: Multi-Objective Optimization

- LP => QP (quadratic programming):

$$\begin{aligned} &max_X\ G(X) \\ &s.t.\ T(X) \geq \alpha \cdot T(X^*) \\ &x_{ij} \geq 0, \forall\, i, j \\ &\sum_j x_{ij} = 1, \forall\, i \end{aligned}$$

$$\Downarrow$$

$$\begin{aligned} &max_X\ G(X) - \tfrac{1}{2}\gamma \| X - Q \|^2 \\ &s.t.\ T(X) \geq \alpha \cdot T(X^*) \\ &x_{ij} \geq 0, \forall\, i, j \\ &\sum_j x_{ij} = 1, \forall\, i \end{aligned}$$

$$\xrightarrow[\text{KKT condition}]{\text{Lagrangian Multiplier}}$$

$$x_{ij} = const + \frac{1}{c \cdot \gamma} \cdot p_{ij} \cdot (1 + \lambda_g \cdot g_j),\ \ if\ x_{ij} > 0$$

Notation	Meaning
i	Indexes users
j	Indexes stores (restaurants)
$X = \{x_{ij}\}$	Serving plan (variable to be optimized)
x_{ij}	Probability that the serving scheme will recommend j th store to user i
$Q = \{q_{ij}\}$	Uniform serving plan with $q_{ij} = \frac{1}{J}$, where J is the total number of open stores.
p_{ij}	Estimated **conversion rate** of user i on store j
g_j	Estimated **gross booking fee** for store j (can be personalized as g_{ij})

Personalized Diversification

- Captures individual's cuisine and diversity preference.

Algorithm 1 IA-SELECT

Input $k, q, C(q), R(q), C(d), P(c|q), V(d|q,c)$
Output set of documents S

$S = \emptyset$
$\forall c, U(c|q,S) = P(c|q)$
while $|S| < k$ **do**
  for $d \in R(q)$ **do**
    $g(d|q,c,S) \leftarrow \sum_{c \in C(d)} U(c|q,S)V(d|q,c)$
  end for
  $d^* \leftarrow argmax\ g(d|q,c,S)$ [ties broken arbitrarily]
  $S \leftarrow S \cup \{d^*\}$
  $\forall c \in C(d^*), U(c|q,S) = (1 - V(d^*|q,c))U(c|q,S \setminus \{d^*\})$
  $R(q) \leftarrow R(q) \setminus \{d^*\}$
end while
return S

Prob(user will order from store | user will not order from all previous recommended stores)

Prob(user will order from cuisine | user will not order from all previous recommended stores)

HRank: Methodology

- A personalized, holistic approach for optimal home feed layout
- Key model components / assumptions
  - Triplet model: ***(eater, store, source)***
  - Scrolling discounting factor
    - User has limited patience
    - Within-row
    - Across-row
  - **Law of total probability**:
    Holistically optimize for **session-level** conversion
    (as opposed to **impression-level** conversion)

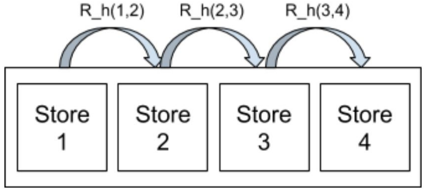


Figure 4: Horizontal impression discounting factor.

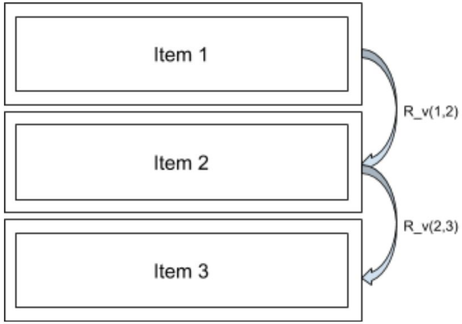


Figure 5: Vertical impression discounting factor.

$$P(O_{ic}=1) = p_1 + (1-p_1)\underline{R_h(1,2)}p_2 + (1-p_1)(1-p_2)\underline{R_h(1,3)}p_3 + \ldots + (1-p_1)\cdots(1-p_{N-1})\underline{R_h(1,N)}p_N.$$

HMOO: Overview

- Challenges
  - MOO only applies to 1-dim world
  - Current world is **2-dim** with heterogeneous & hierarchical contents
- Soln: 2-step procedure
  - 2-dim => 1-dim
    - Define MOO objective on item (restaurant/carousel) level
  - Apply 1-dim MOO
- Goal
  - A personalized, *holistic* home feed that is optimized for all sides of the marketplace
    - HRank: holistic optimization of eater conversion
    - HMOO: holistic optimization of marketplace objectives (e.g. gross bookings)

HMOO: Methodology

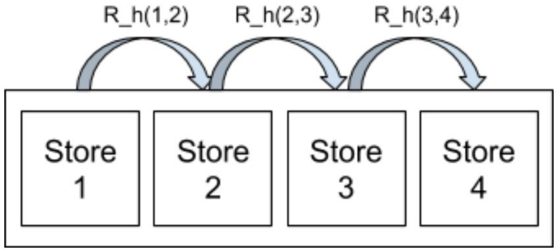


Figure 4: Horizontal impression discounting factor.

- Expanding the HRank framework
  - Law of total probability => HRank
  - Law of total expectation => HMOO

$$E[B_{ic}] = E[B_{ic} \mid O_{ic} = 1]P(O_{ic} = 1) + E[B_{ic} \mid O_{ic} = 0]P(O_{ic} = 0).$$

Expected $ from a carousel

Conversion rate of the carousel

HMOO: Methodology

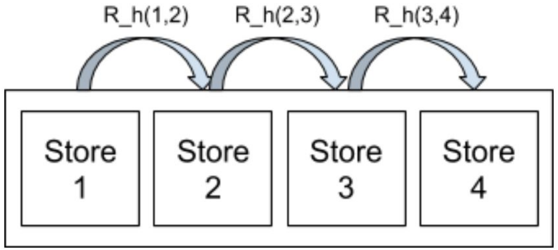


Figure 4: Horizontal impression discounting factor.

- Expanding the HRank framework
  - Law of total probability => HRank
  - Law of total expectation => HMOO

$$E[B_{ic}] = \underline{E[B_{ic} \mid O_{ic} = 1]}P(O_{ic} = 1) + \cancel{E[B_{ic} \mid O_{ic} = 0]P(O_{ic} = 0)}.$$

Expected $ from a carousel

By Law of total expectation

HMOO score of the carousel

Conversion rate of the carousel

HRank

  - Readily applies to **other objectives / multiple objectives**

HMOO

Basket size objective for carousels

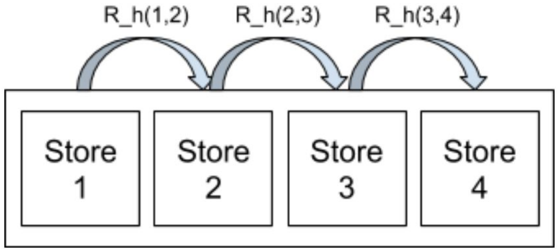


Figure 4: Horizontal impression discounting factor.

- Law of total expectation:

$$E[B_{ic}] = E[B_{ic} \mid O_{ic} = 1]P(O_{ic} = 1) + \cancel{E[B_{ic} \mid O_{ic} = 0]P(O_{ic} = 0)}.$$

$$E[B_{ic}] = p_1 b_1 + (1-p_1)R_h(1,2)p_2 b_2 + (1-p_1)(1-p_2)R_h(1,3)p_3 b_3 + \ldots + (1-p_1)\cdots(1-p_{N-1})R_h(1,N)p_N b_N.$$

$$P(O_{ic} = 1) = p_1 + (1-p_1)R_h(1,2)p_2 + (1-p_1)(1-p_2)R_h(1,3)p_3 + \ldots + (1-p_1)\cdots(1-p_{N-1})R_h(1,N)p_N.$$

- Putting everything together

$$E[B_{ic} \mid O_{ic} = 1] = \frac{p_1 b_1 + (1-p_1)R_h(1,2)p_2 b_2 + (1-p_1)(1-p_2)R_h(1,3)p_3 b_3 + \ldots + (1-p_1)\cdots(1-p_{N-1})R_h(1,N)p_N b_N}{p_1 + (1-p_1)R_h(1,2)p_2 + (1-p_1)(1-p_2)R_h(1,3)p_3 + \ldots + (1-p_1)\cdots(1-p_{N-1})R_h(1,N)p_N}$$