UBER EATS

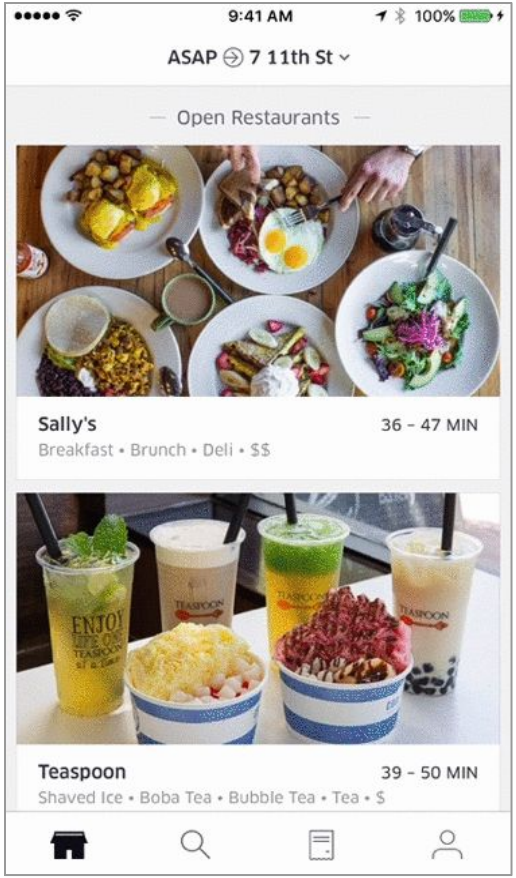
UberEATS Restaurant Ranking & Recommendation

EATS Mission

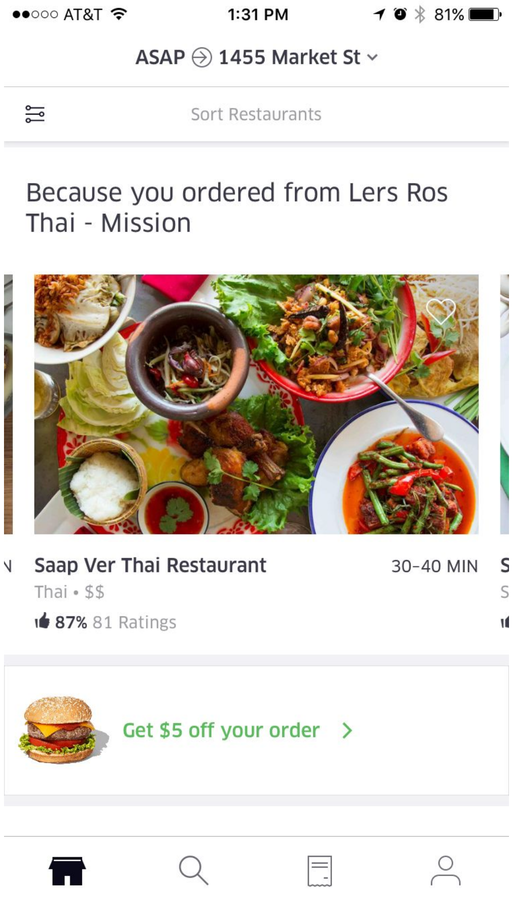
Make eating well effortless at any time, for anyone

Background

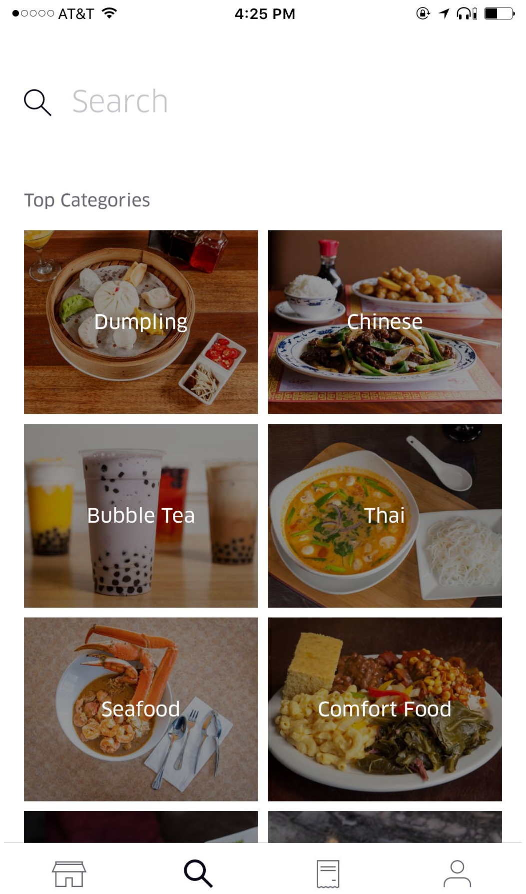
- **Goal of Ranking and Recommendation**
  - Recommend each user the most relevant restaurants, at the right time, with the right context
- **Similar problems**
  - Netflix movie recommendation
  - LinkedIn job recommendation
- **Challenges**
  - Personalization
  - Help the new/low volume restaurants
  - Multiple factors (eater, restaurant, couriers, etc.)
  - Understand the food
  - ...



Main Feed



Carousels

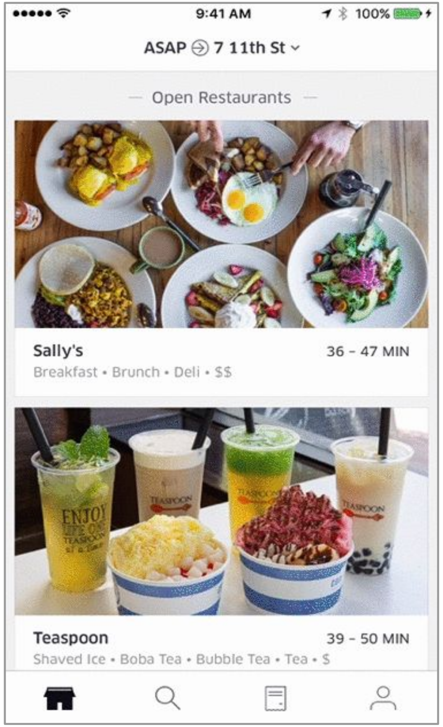


Search

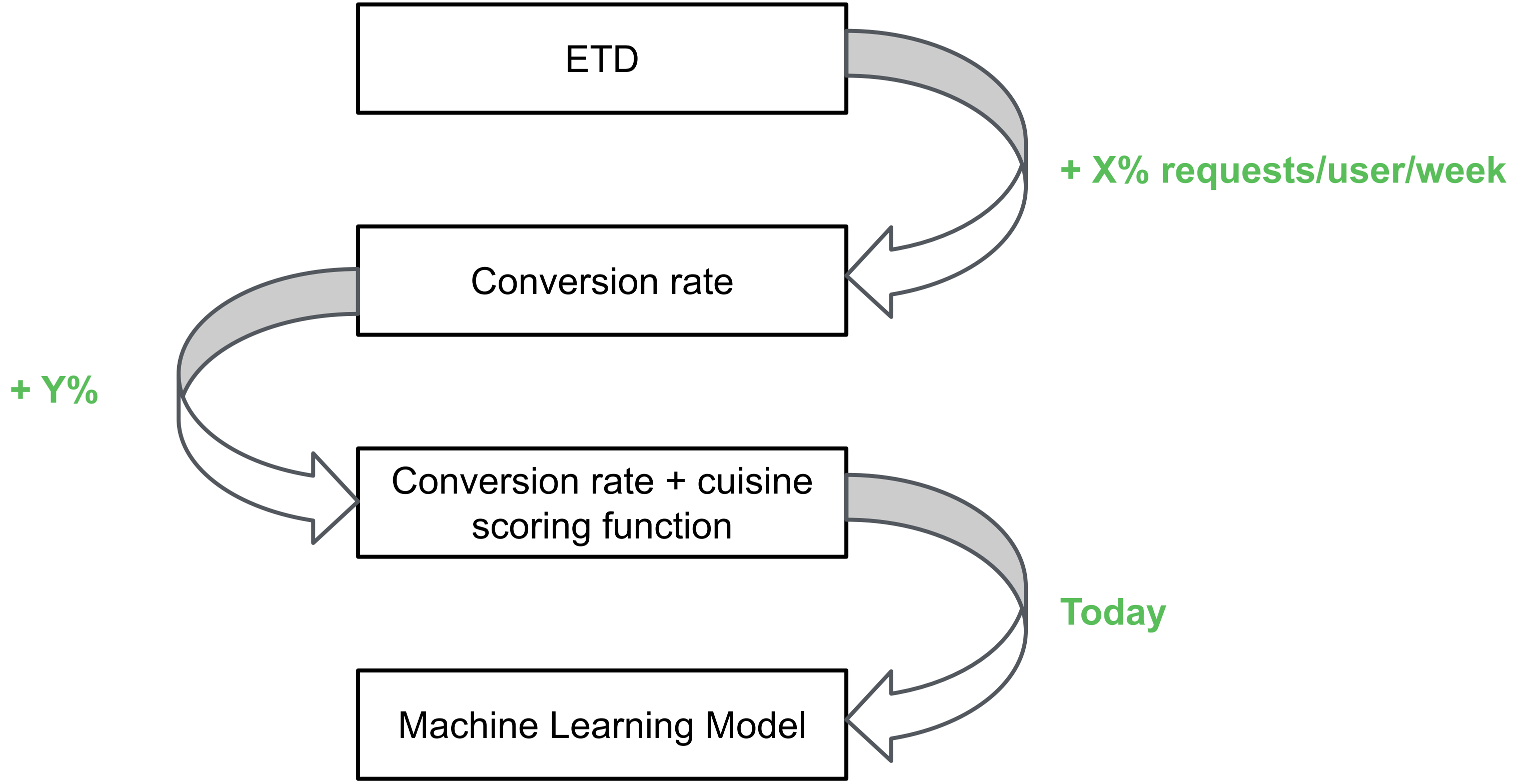
Outline

Ranking for the Main Feed

- Ranker Evolution
- Multi-Objective Optimization (MOO)
- Bayesian Bandits to help new restaurants
- Holistic Optimization: session level models & diversification



EATS Ranker Evolution

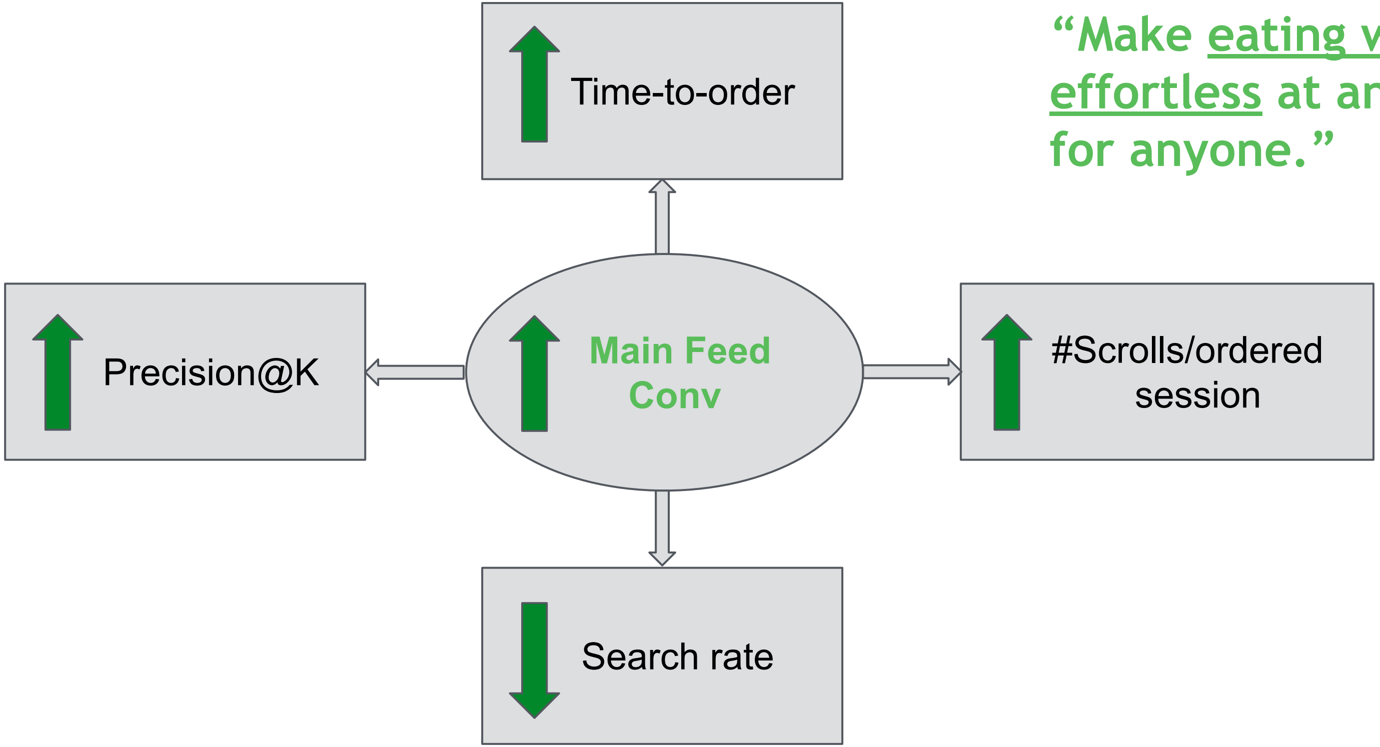


Main Feed Ranking Pipeline

- **Real-time**
  - For every *(eater, store)* pair
- **Personalized**
  - Based on a large pool of eater, restaurant and contextual features
    - Restaurant features
    - Eater features
    - Eater-restaurant interaction features
    - Contextual features
    - Collaborative filtering features
- **Predictive modeling**
  - ~10 million rows/day
  - Michelangelo for model training and online model serving

Online performance: other metrics

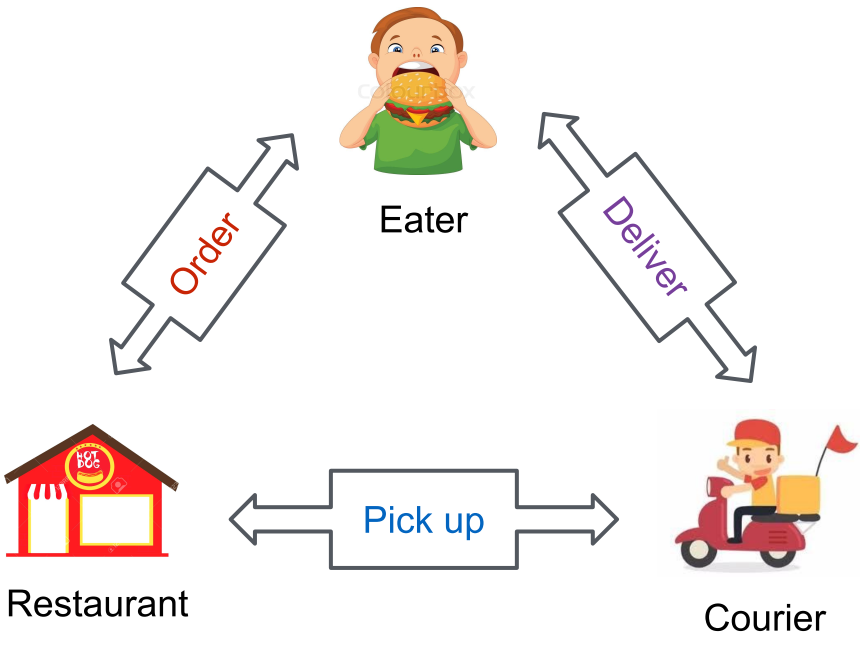
Better user experience from multiple perspectives



"Make <u>eating well effortless</u> at any time, for anyone."

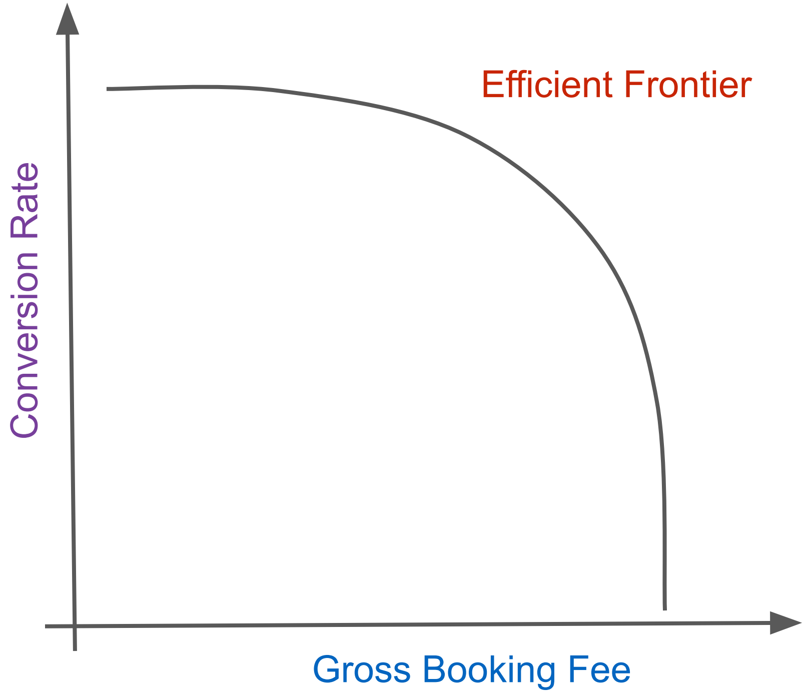
MOO: Multi-Objective Optimization

- Motivation
  - [Before] Single objective: optimized for (impression level) conversion rate
  - UberEATS is a unique 3-sided marketplace
    - Eater: satisfaction / retention
    - Restaurant: happiness / fairness
    - Courier / marketplace health: supply-demand efficiency
    - Uber: gross booking fee / net inflow
    - ...



MOO: Multi-Objective Optimization

- Motivation
  - [Before] Single objective: optimized for (impression level) conversion rate
  - UberEATS is a unique 3-sided marketplace
    - Eater: satisfaction / retention
    - Restaurant: happiness / fairness
    - Courier / marketplace health: supply-demand efficiency
    - Uber: gross booking fee / net inflow
    - ...



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):

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

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):

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

$$\Downarrow$$

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

Lagrangian Multiplier

$$\Longrightarrow$$

KKT condition

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

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})

MOO: Multi-Objective Optimization

- Solution:

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

- Ranking according to x_{ij} is **equivalent** to ranking according to:

$$s_{ij} = p_{ij} \cdot (1 + \lambda_g \cdot g_j)$$

- ❑ A nice, simple and intuitive formula that does the trade-off among different objectives.
- ❑ Can be easily extended to multiple objectives:

$$s_{ij} = p_{ij} \cdot (1 + \lambda_g \cdot g_j + \lambda_r \cdot r_j + \lambda_{rt} \cdot rt_j)$$

- ❑ Also allows personalized objectives:

$$s_{ij} = p_{ij} \cdot (1 + \lambda_g \cdot g_{ij} + \lambda_r \cdot r_{ij} + \lambda_{rt} \cdot rt_{ij})$$

Bayesian Bandit

- Motivation: **Explore/exploit tradeoff**
  - Exploit
    - Restaurants with high predicted conversion rate
  - Explore
    - New / low volume restaurants
  - Bayesian bandit
    - Bayesian modeling
    - Contextual multi-armed bandit

Bayesian Bandit

- Bayesian modeling
  - Prior distribution for conversion rate

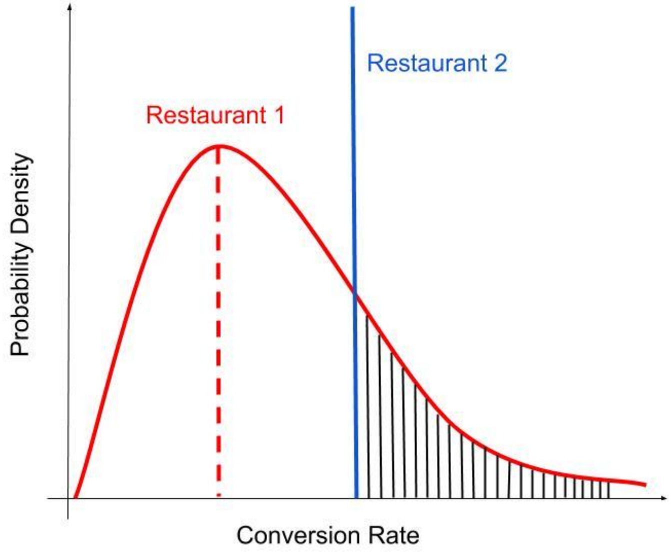
    $$p_j \sim Beta(\alpha, \beta) \text{ for restaurant } j$$

  - **Posterior variance**

    $$\sigma_j^{\,2} = \frac{\hat{p}_j(1-\hat{p}_j)}{\alpha+\beta+N_j+1}$$

    Where N_j is number of impressions j receives.

  - New restaurant - High variance
  - Well-established restaurant - Low variance



Bayesian Bandit (delete this slide)

- Contextual multi-armed bandit
  - Conversion rate model to get

$$\hat{p}_{ij} = f(X_{user}, X_{store}, X_{user-store}, X_{contextual}, X_{CL})$$

  - **UCB** for explore-exploit:

$$s_j = \hat{p}_{ij} + \varkappa \cdot \sigma_j$$

- Combined with other MOO objectives:

$$s_{ij} = p_{ij} \cdot (1 + \lambda_g \cdot g_{ij} + \lambda_r \cdot r_{ij} + \lambda_{rt} \cdot rt_{ij}) + \varkappa \cdot \sigma_j$$

- Online performance
  - **+150%** in impression percentage on new/low-vol restaurants
  - **+100%** in order percentage on new/low-vol restaurants
  - No drop in conversion rate, rpu, retention

Before: Tradeoff

How do you make the tradeoff?

Suppose we have 10 objectives, each with weight; How to find the optimal weight combination?

-- Bayesian optimation

Train a model based on the response -- black box model (we don’t assume any functional form) -- optimize it -- where Bayesian optimization into play

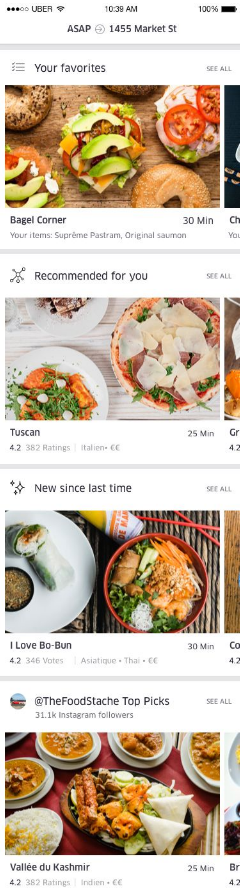
(don’t mention unless asked: Ultimate objective: LTV of all the partners in the marketplace)

Holistic Optimization (need advice here)

- Motivation
  - Current ranking framework is **myopic**: optimizing conversion rate / MOO score **independently** at each position / impression;
  - Ultimate goal of ranking: learn a personalized app homepage that is optimized for the *session*-level conversion rate
  - *(user, store)* -> *(user, session)*?

Holistic Optimization: Session level model

- Session-Level Model
  - Takes in session-level features and predicts conversion for every *(user, session)* pair
  - Features:
    - # of carousels shown
    - boosting factors for MOO objectives
    - summary statistics of predicted conversion rate for individual items
    - contextual features
    - …
- Use case of the session level model:
  - Optimal # of carousels shown in each session
  - Optimal boosting factors



Holistic Optimization: Session level model

- [Bayesian Optimization with Contextual Multi-Armed Bandit]()
  - Model training: Learn the model through ML or **Gaussian Process**:

$$y_{ij} = f(\theta, x_{ij}) + \varepsilon_{ij}$$

  - Online policy optimization:
    - Random Search

$$\theta_{n+1} \mid x_{n+1} = argmax_{\theta \in \Theta}\, f(\theta; x_{n+1})$$

    - Contextual Bayesian Optimization with MAB

$$a_{UCB}(\theta;\, x_{n+1}) = m_n(\theta;\, x_{n+1}) + \varkappa \cdot \sigma_n(\theta;\, x_{n+1}),$$

$$\theta_{n+1} \mid x_{n+1} = argmax_{\theta \in \Theta}\, a_{UCB}(\theta;\, x_{n+1}),$$

Holistic Optimization: Diversification

- Motivation
  - Current ranking framework is **myopic**: optimizing conversion rate / MOO score **independently** at each position / impression;
  - Potentially could cause negative user experience:
    - "McDonald's everywhere"
    - Overwhelming restaurants that are too similar to each other in a consecutive order.

- Diversification algorithm
  - (V1) Non-personalized: maximal marginal relevance (MMR);
  - (V2) Personalized: captures individual's cuisine and diversity preference.

Holistic Optimization: Diversification V1

- **V1 [Non-personalized]**: maximal marginal relevance (MMR)

$$MMR(i,j) := argmax_{s_j \in S^c}[\lambda \cdot R_{ij} - (1-\lambda) \cdot max_{s_k \in S} sim(s_k, s_j)]$$

where

$$sim(s_k, s_j) := V(s_k)^T V(s_j) / (|V(s_k)|_2 \cdot |V(s_j)|_2)$$

and $V(\cdot)$ is the **vector representation** (stay tuned) of a store / carousel.

Holistic Optimization: Diversification V2

- **V2 [Personalized]**: Captures individual's cuisine and diversity preference.

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)

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 $|\mathrm{S}| < \mathrm{k}$ **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

Summary

The more questions you got, the better the presentation

Guideline:

Focus more on why instead of how

importance of personalized recommendation

remove numbers of the lifts

no second pass ranker

why for MOO
talk about 3 sided marketplace
balance between 3 sides

Holistic ranking:
introduce carousels (ask Zhen for the Benu slides)
optimize for the whole homepage

Carousel: borrow narrative on rows from [netflix blog]

Questions

1. Shall we use the same slide template