

# SISTEMAS DE RECOMENDACIÓN (ENCORE)

**Cristian Cardellino - Luis Biedma**

Uber Eats

PRIMERO, UNA CURIOSIDAD...



# QUÉ QUIERO COMER?

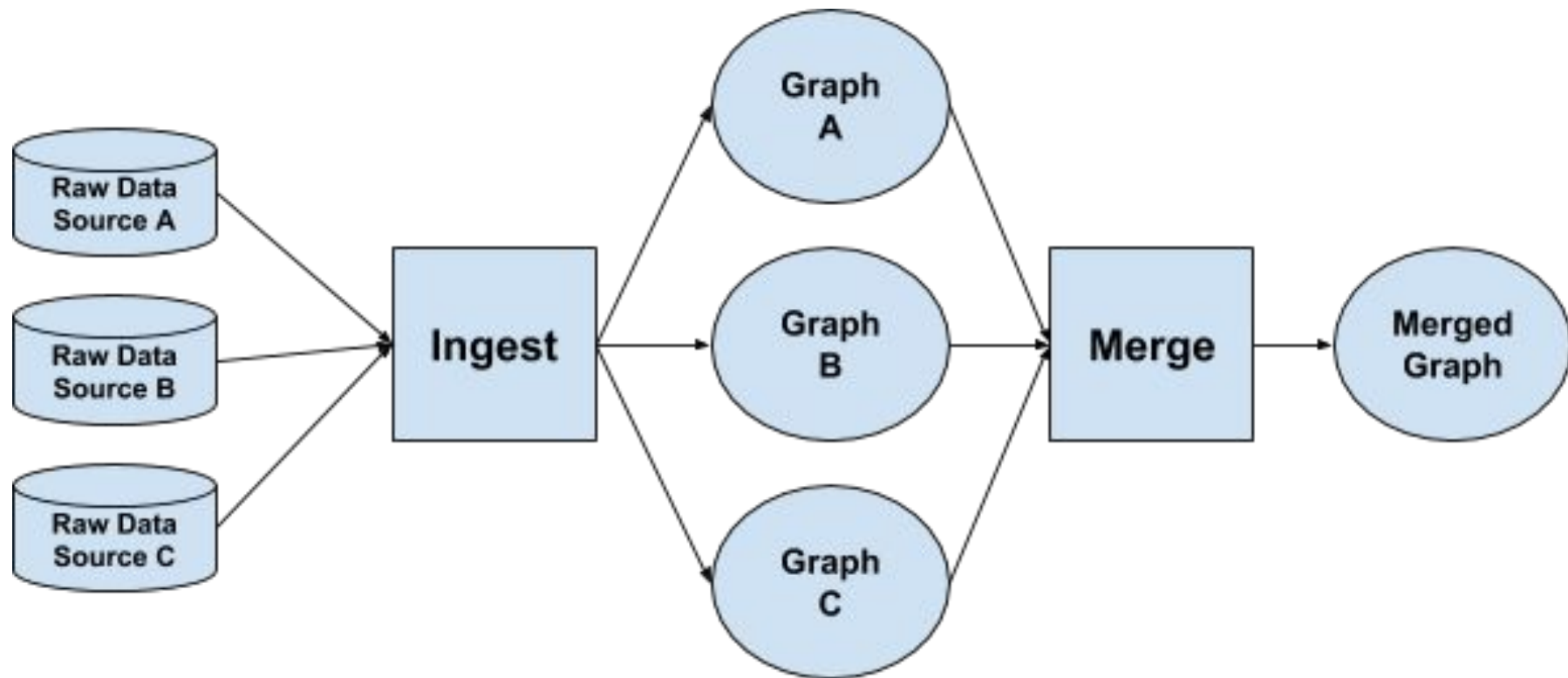
- A veces, sé que voy a entrar a la app a pedir que me manden una bolsa de criollitos
- Otras veces, entro para ver qué quiero almorzar y no tengo idea.

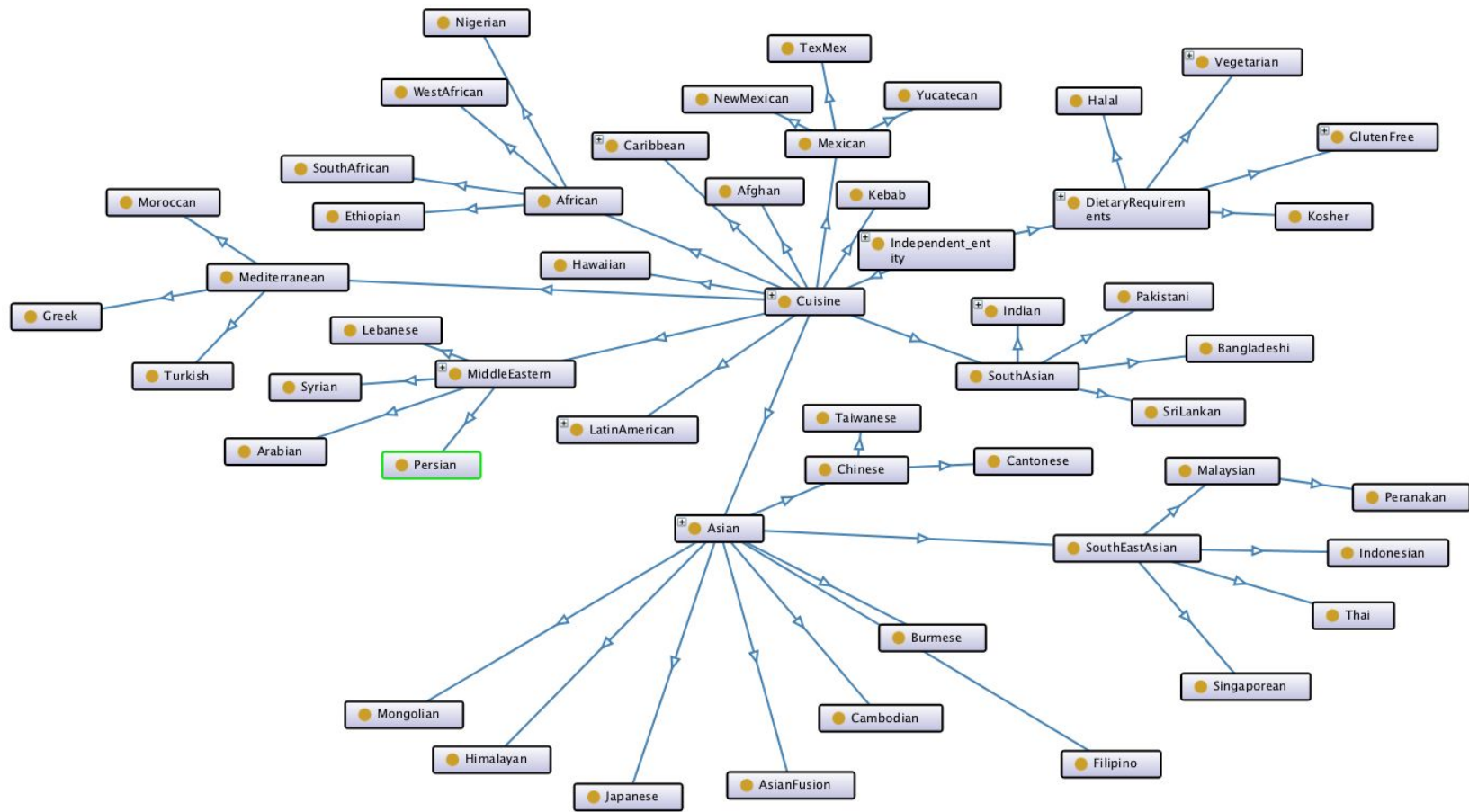


# GRAFO DE CONOCIMIENTO CULINARIO

- Viene de Query Understanding...
- ... pero la intención es “ambigua”
- Para eso hay que desarrollar FER (Food-Entity Recognition) con una Knowledge Base propia
- Restaurantes, Cocinas, Menús...
- Podemos hacer un grafo!

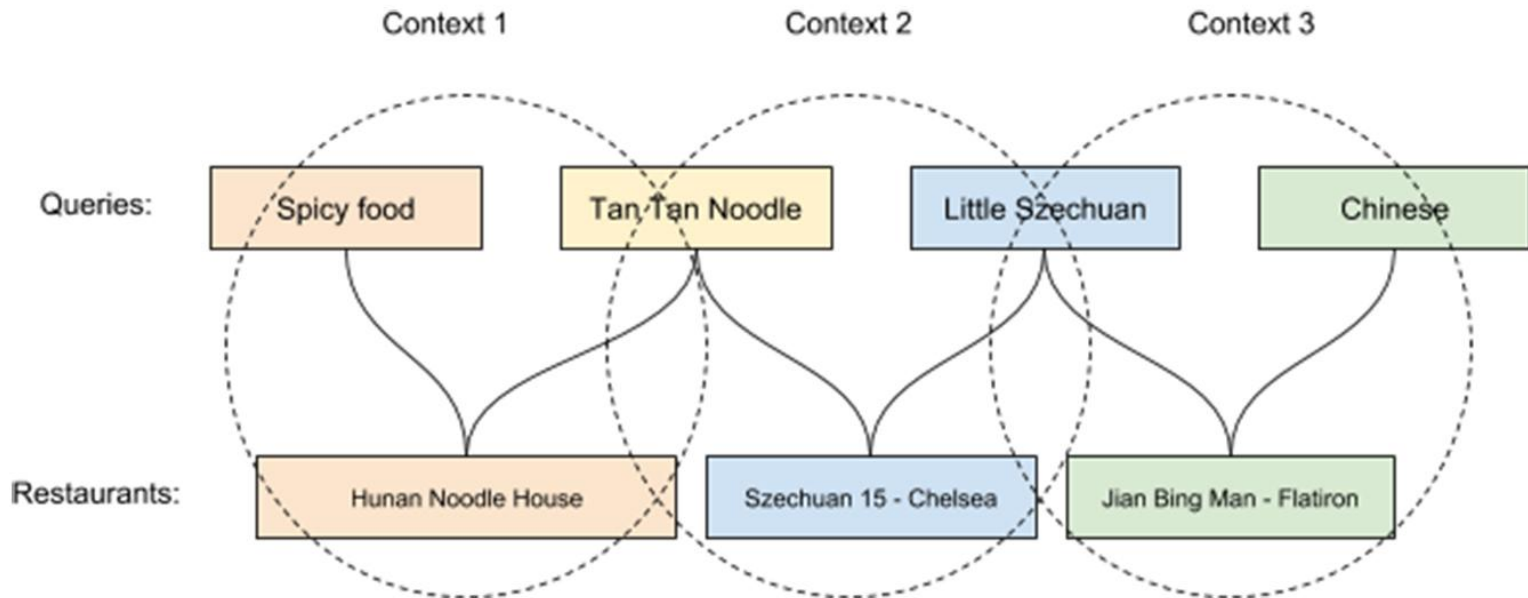
# UN POQUITO DE ESTRUCTURA





# RESUELVE UN PAR DE PROBLEMAS

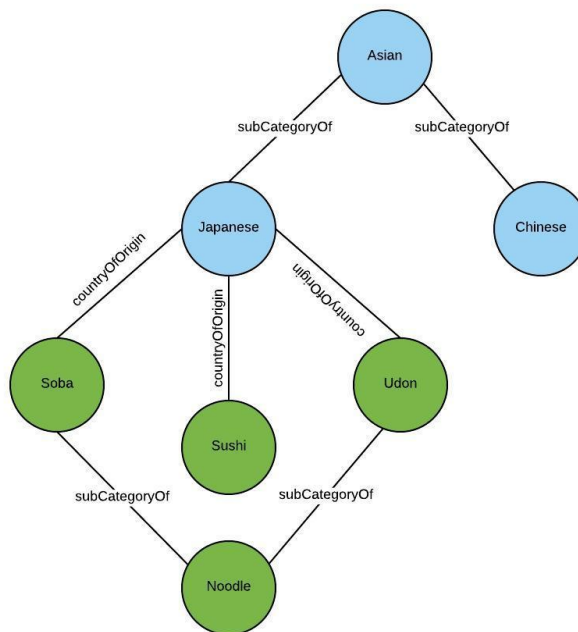
Food Discovery: Query2Vec





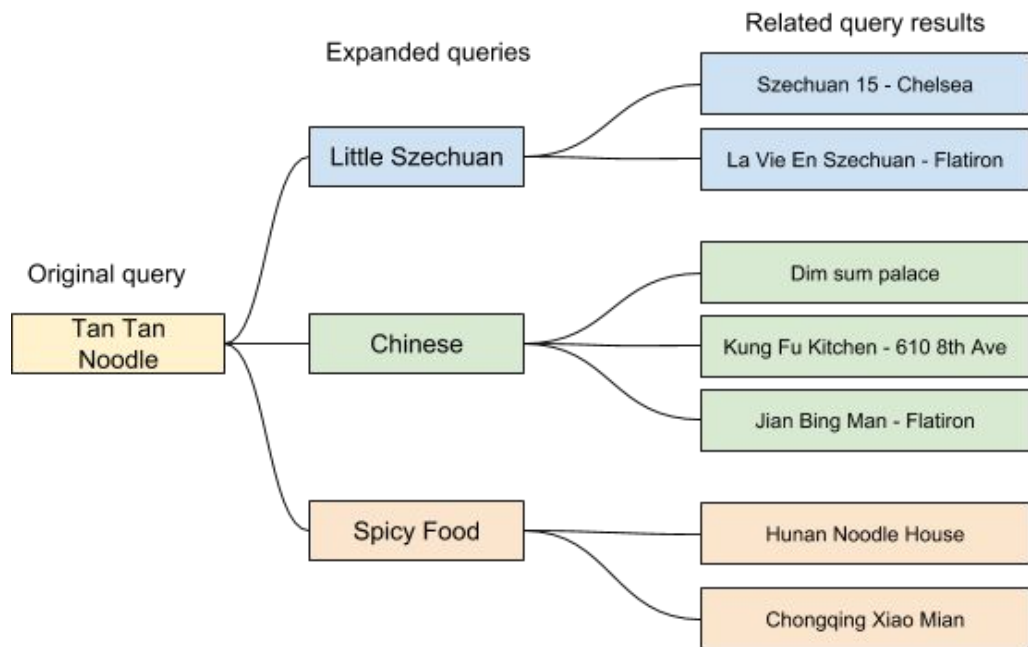
# RESUELVE UN PAR DE PROBLEMAS

- Zero Result Problem

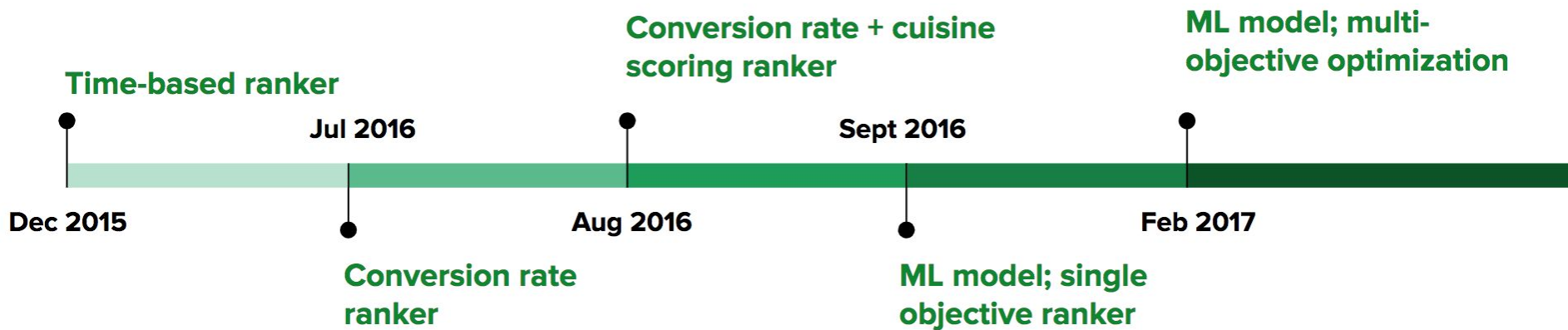


# RESUELVE UN PAR DE PROBLEMAS

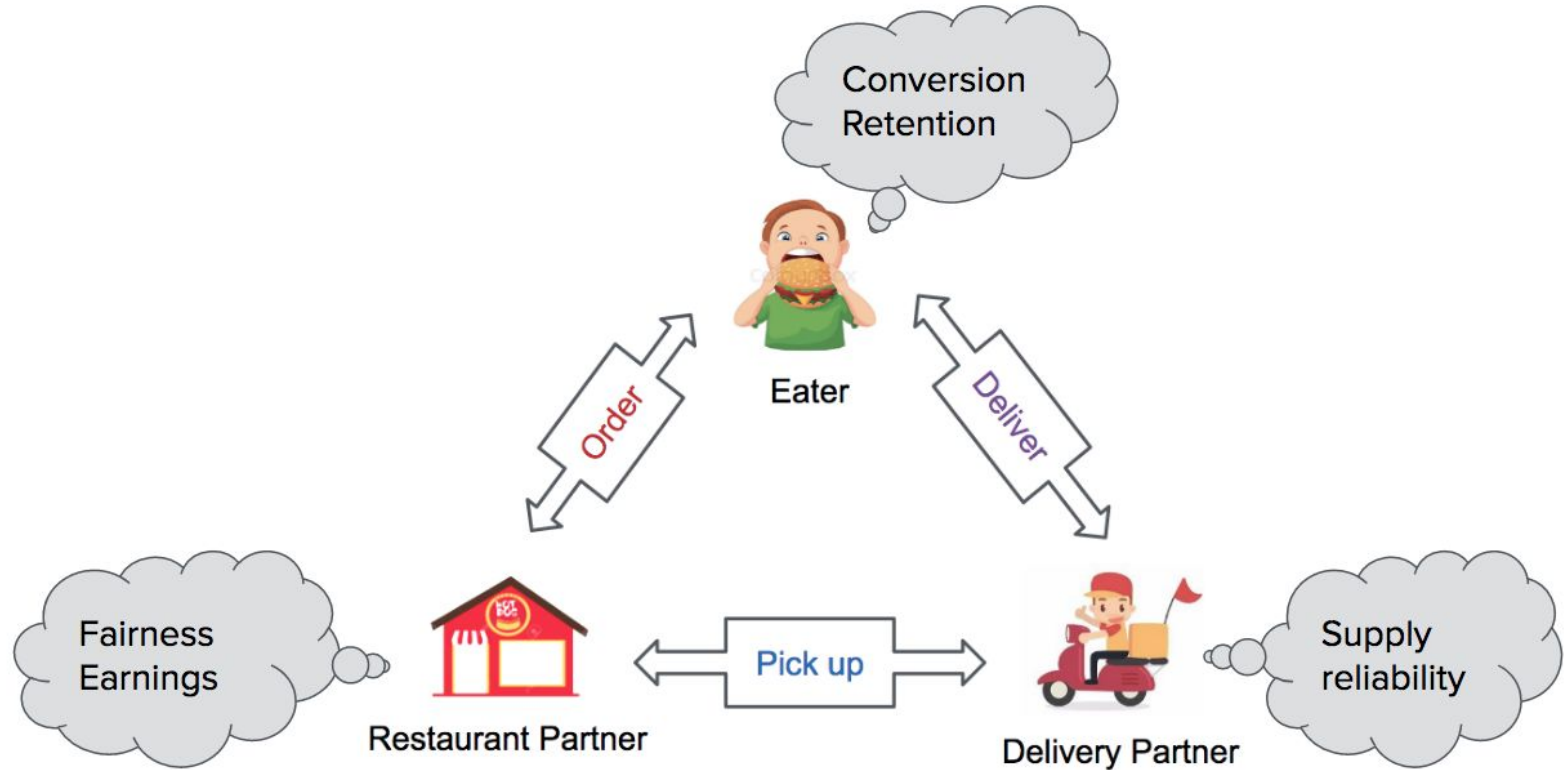
- Zero Result Problem: Query Expansion + Learning to Rank



# EVOLUCIÓN DEL RECOMENDADOR DE UBEREATS



# ECOSISTEMA DE UBEREATS



# ECOSISTEMA DE UBEREATS

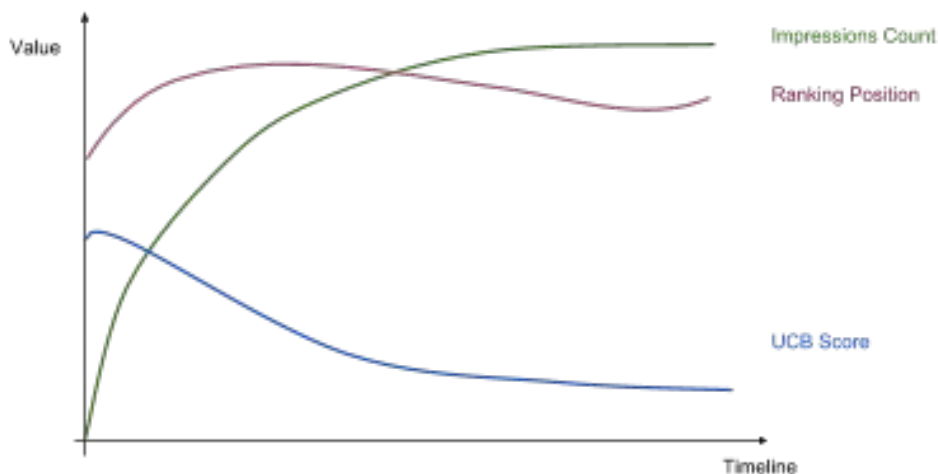
## Comensales

- Parte crucial, es la que permite escalar
- Receptores del Sistema de Recomendación
- Varias cosas a resolver
  - Similaridades
  - Diversidad
  - Retención

# ECOSISTEMA DE UBEREATS

## Restaurantes

- Popularidad dependiente de comensales
- Boost de restaurantes “no tan conocidos”: Multi-Armed Bandit



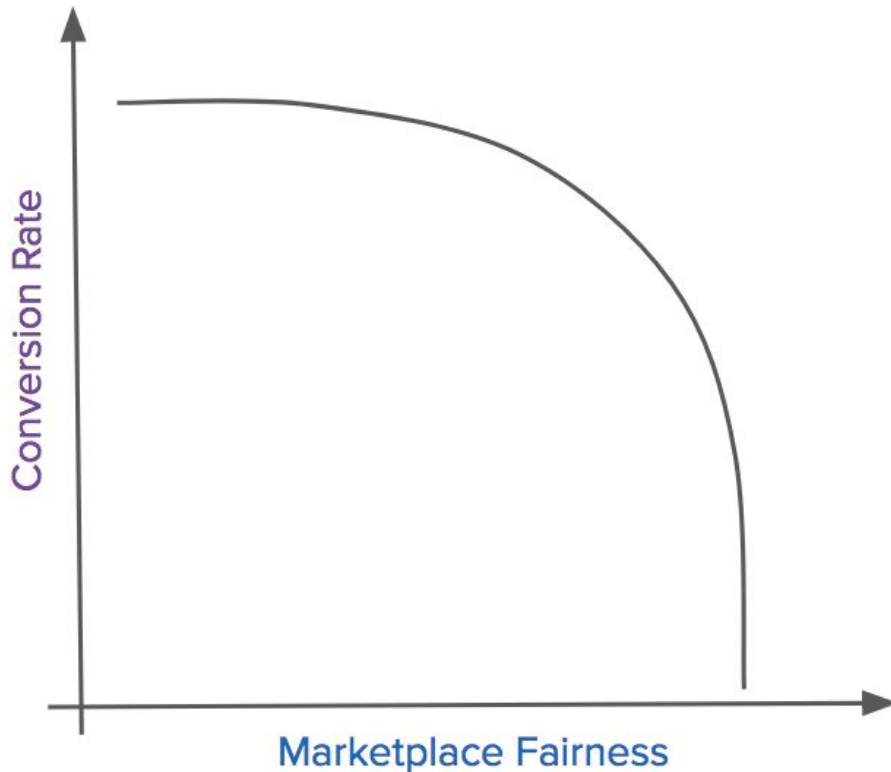
# ECOSISTEMA DE UBEREATS

## Delivery Partners

- Es necesario darles un “buen camino”
  - <https://eng.uber.com/uber-eats-trip-optimization/>
- Motivados por la ganancia
  - Tiempos de espera más cortos
  - Caminos más cortos
  - [Michelangelo](#)
- Un restaurant que tarda mucho en darle la comida al delivery puede hacer bajar su recomendación

# OPTIMIZACIÓN MULTI-OBJETIVO

- Poca correlación entre objetivos distintos
- Principio matemático que tiene en cuenta trade-offs
- [Inserte fórmula un poco densa]
- Luego, aplica Learning to Rank + Explore/Exploit





# RECOMMENDING FOR THE MARKETPLACE

