Building a Real-Time Fraud Prevention Engine Using Open Source (Big Data) Software

Kees Jan de Vries
Booking.com

SPARK SUMMIT EAST 2017

Who am I?

About me

- Physics PhD Imperial College
- Data Scientist at Booking.com for 1.5 year
  - Security Department
- linkedin.com/in/kees-jan-de-vries-93767240

About Booking.com

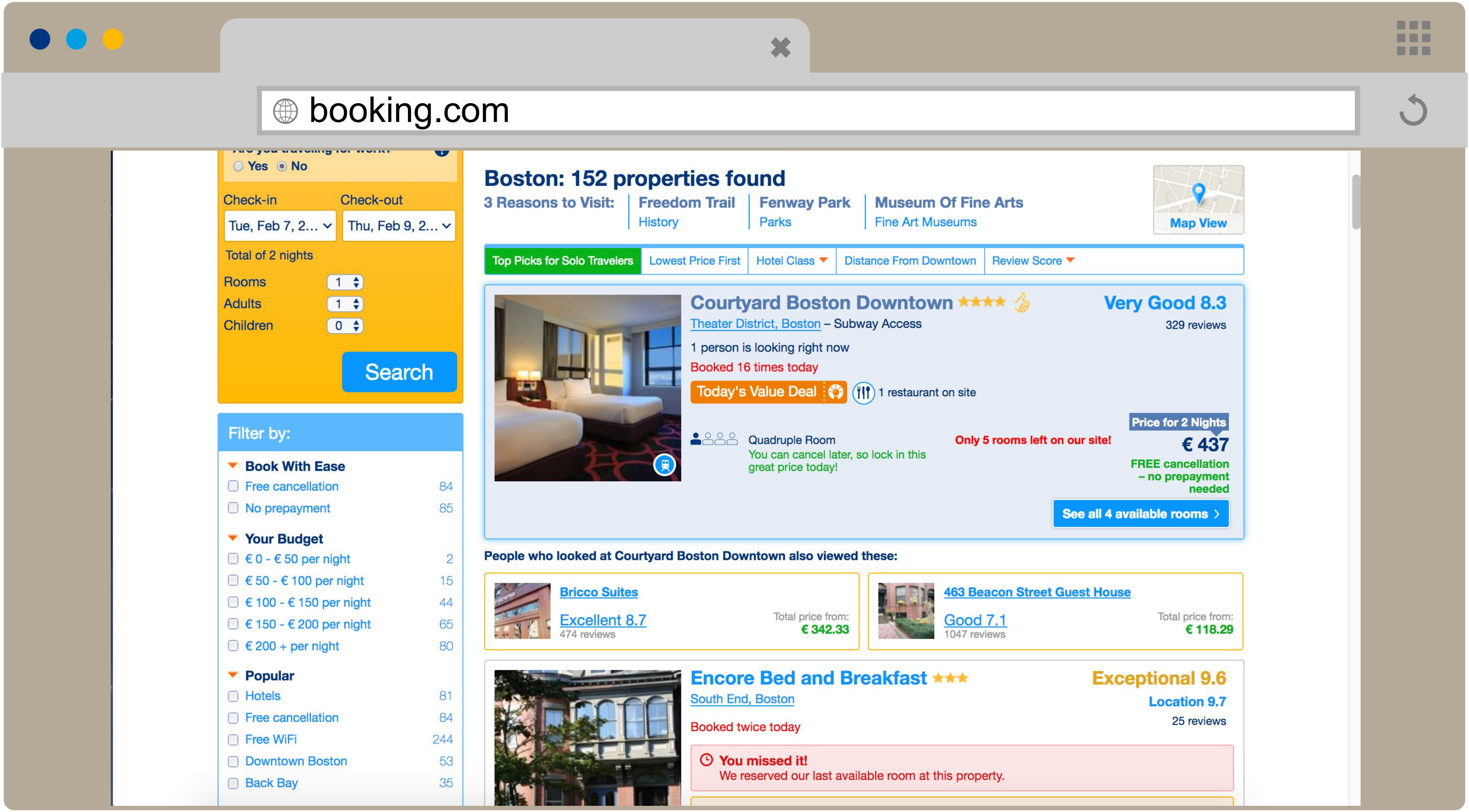
- World leader in connecting travellers with the widest variety of great places to stay
- Part of The Priceline Group, the world's 3rd largest e-commerce company (by market capitalisation)
- Employing 14,000 people in 180 countries
- Each day, over 1,200,000 room nights are reserved on Booking.com

Contents.

- Motivation
- Running Example
  - Probability to Book
- Real Time Prediction Engine: Lessons Learnt
  - Aggregate Features
  - Models Training and Deployment
  - Interpretation of Individual Scores

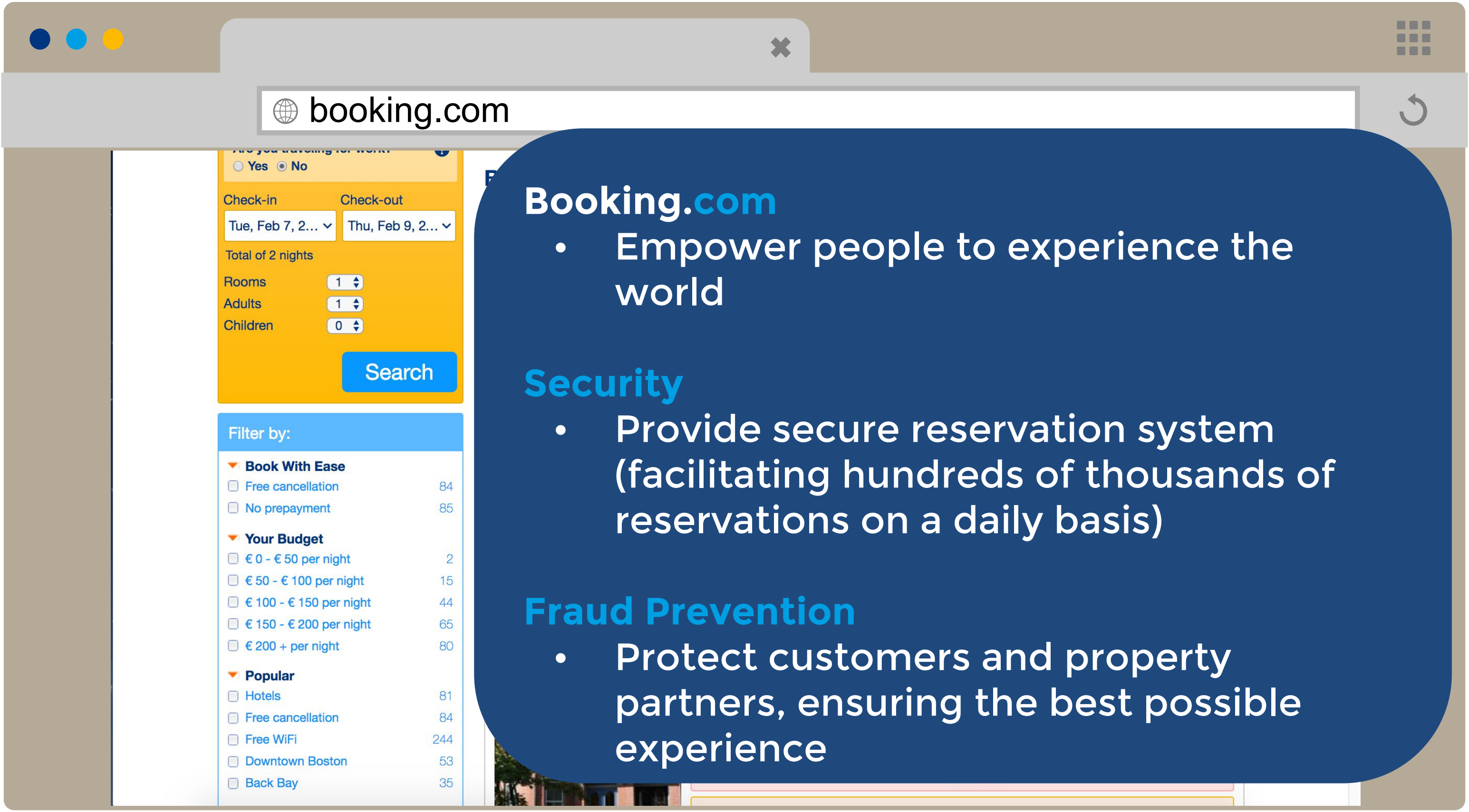
Motivation

Motivation.



booking.com
Are you traveling for work?
Yes No
Check-in
Check-out
Tue, Feb 7, 2...
Thu, Feb 9, 2...
Total of 2 nights
Rooms 1
Adults 1
Children 0
Search
Filter by:
Book With Ease
Free cancellation 84
No prepayment 85
Your Budget
€ 0 - € 50 per night 2
€ 50 - € 100 per night 15
€ 100 - € 150 per night 44
€ 150 - € 200 per night 65
€ 200 + per night 80
Popular
Hotels 81
Free cancellation 84
Free WiFi 244
Downtown Boston 53
Back Bay 35
Boston: 152 properties found
3 Reasons to Visit:
Freedom Trail
History
Fenway Park
Parks
Museum Of Fine Arts
Fine Art Museums
Map View
Top Picks for Solo Travelers
Lowest Price First
Hotel Class
Distance From Downtown
Review Score
Courtyard Boston Downtown
Very Good 8.3
329 reviews
Theater District, Boston – Subway Access
1 person is looking right now
Booked 16 times today
Today's Value Deal
1 restaurant on site
Price for 2 Nights
Quadruple Room
You can cancel later, so lock in this great price today!
Only 5 rooms left on our site!
€ 437
FREE cancellation – no prepayment needed
See all 4 available rooms
People who looked at Courtyard Boston Downtown also viewed these:
Bricco Suites
Excellent 8.7
474 reviews
Total price from:
€ 342.33
463 Beacon Street Guest House
Good 7.1
1047 reviews
Total price from:
€ 118.29
Encore Bed and Breakfast
Exceptional 9.6
South End, Boston
Location 9.7
25 reviews
Booked twice today
You missed it!
We reserved our last available room at this property.

Motivation.



Booking.com

- Empower people to experience the world

Security

- Provide secure reservation system (facilitating hundreds of thousands of reservations on a daily basis)

Fraud Prevention

- Protect customers and property partners, ensuring the best possible experience

Motivation for this Talk.

Train awesome Model

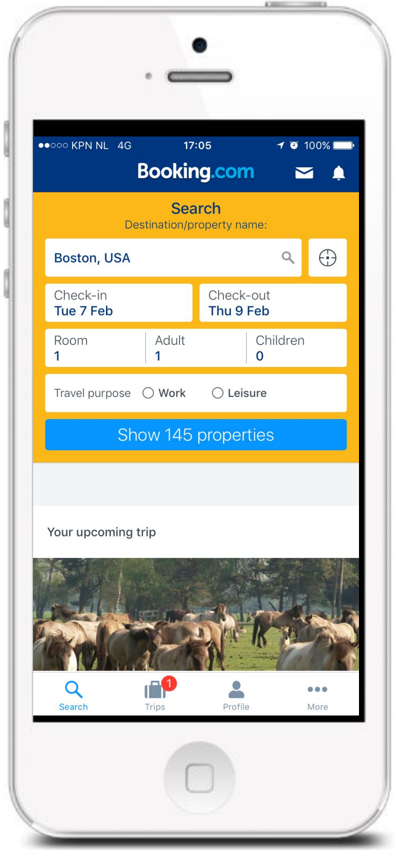
Serve in Real Time
at Low Latency using
aggregate features

Running Example

Probability to Book

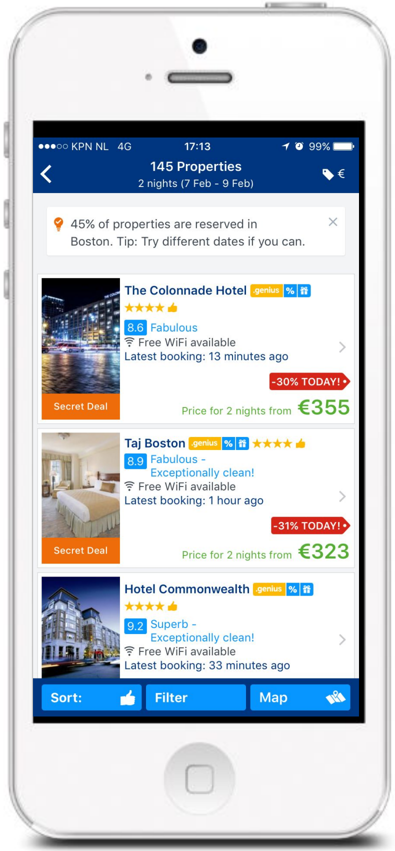
Disclaimer: although the running example presented in these slides may seem realistic, it is only intended to highlight some lessons learnt about building a real time prediction engine.

Probability to Book.



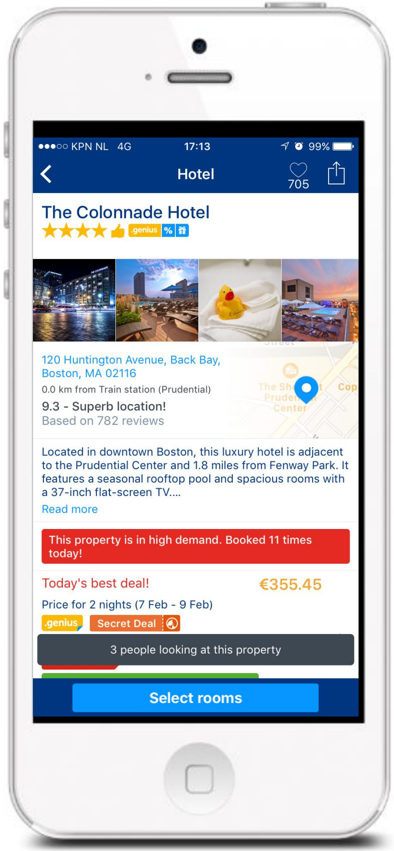
Let’s book a hotel for the Spark Summit East 2017 in Boston

Probability to Book.



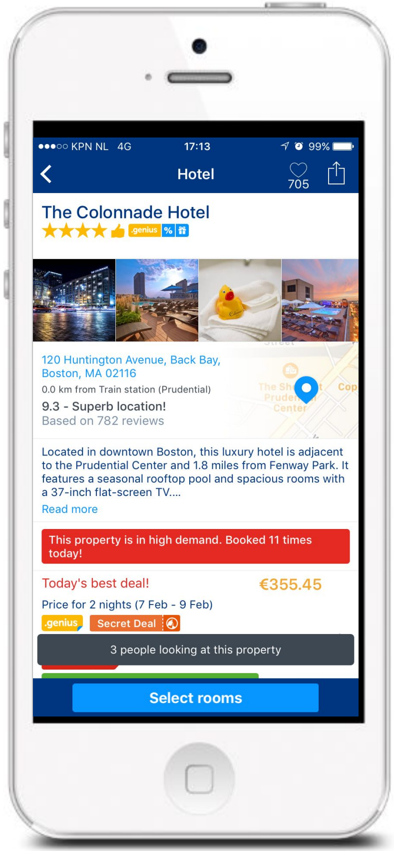
OK. Let's click on the first hit.

Probability to Book.



Nice. Here's all the information I need. But maybe I'll browse a few more, to make the best choice.

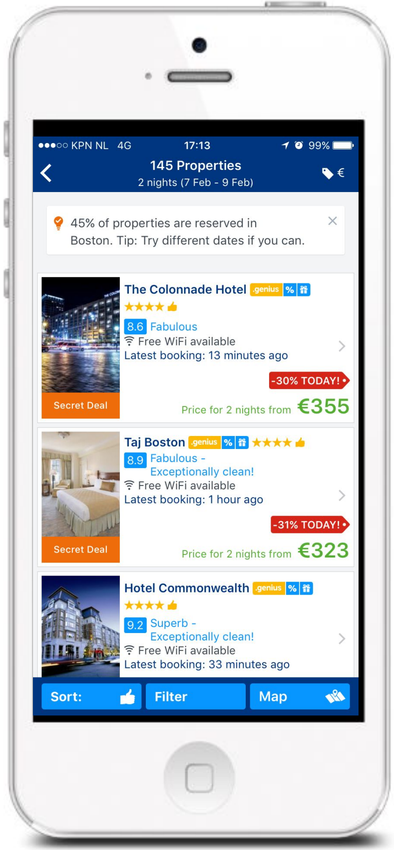
Probability to Book.



Nice. Here's all the information I need. But maybe I'll browse a few more, to make the best choice.

Let's help the customer make the best choice for them!

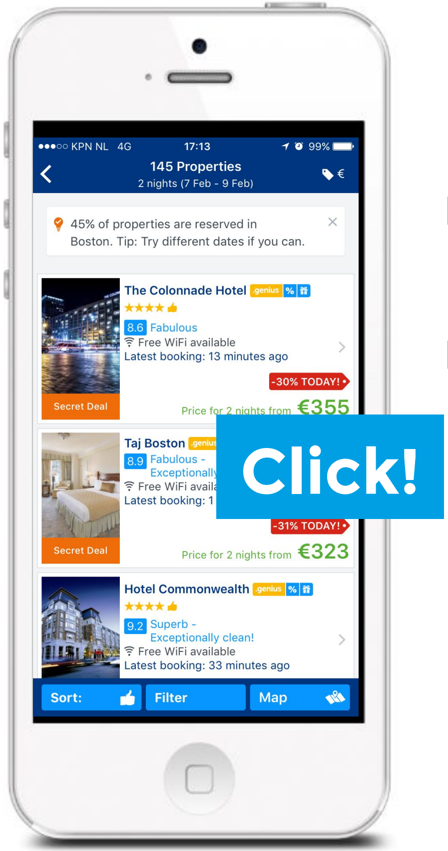
Probability to Book.



Nice. Here's all the information I need. But maybe I'll browse a few more, to make the best choice.

We'll calculate the probability of booking when the Customer clicks on an accommodation

Behind the Scenes.



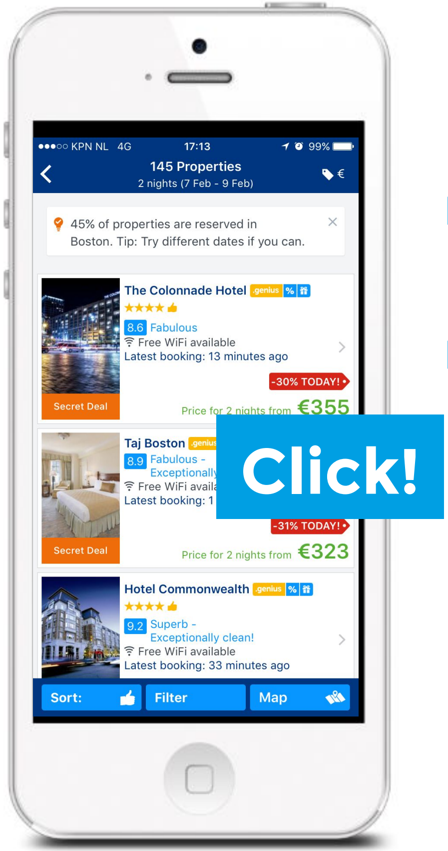
Behind the Scenes.

Labels

- Booked? Yes/No

Features

- Simple
  - Time of day
  - ...
- Profile
  - User
  - Circumstantial



Behind the Scenes.

Labels

- Booked? Yes/No

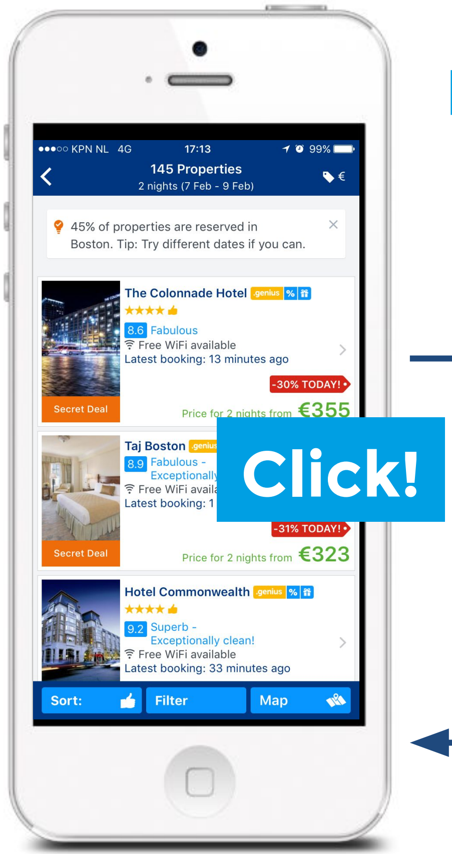
Features

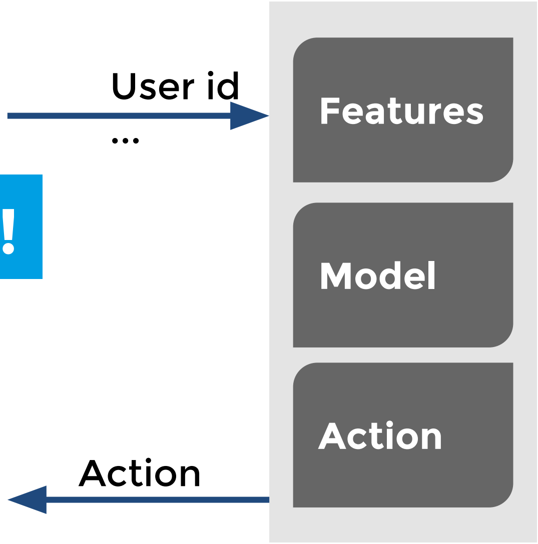
- Simple
  - Time of day
  - ...
- Profile
  - User
  - Circumstantial

Aggregate

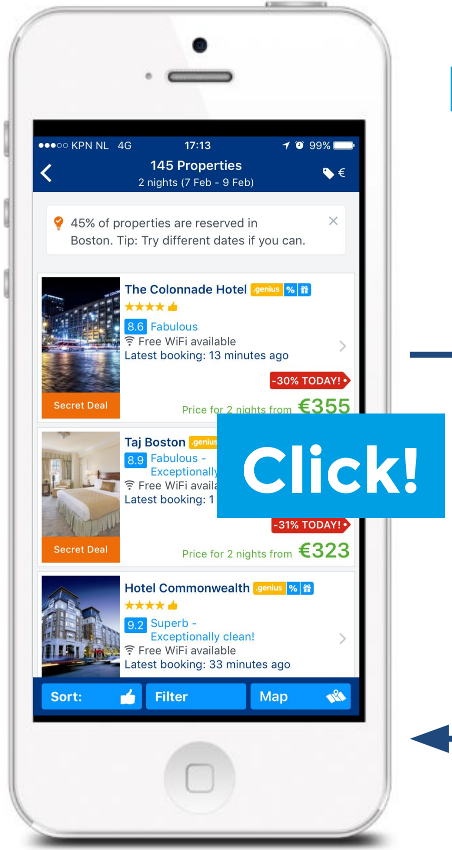
- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

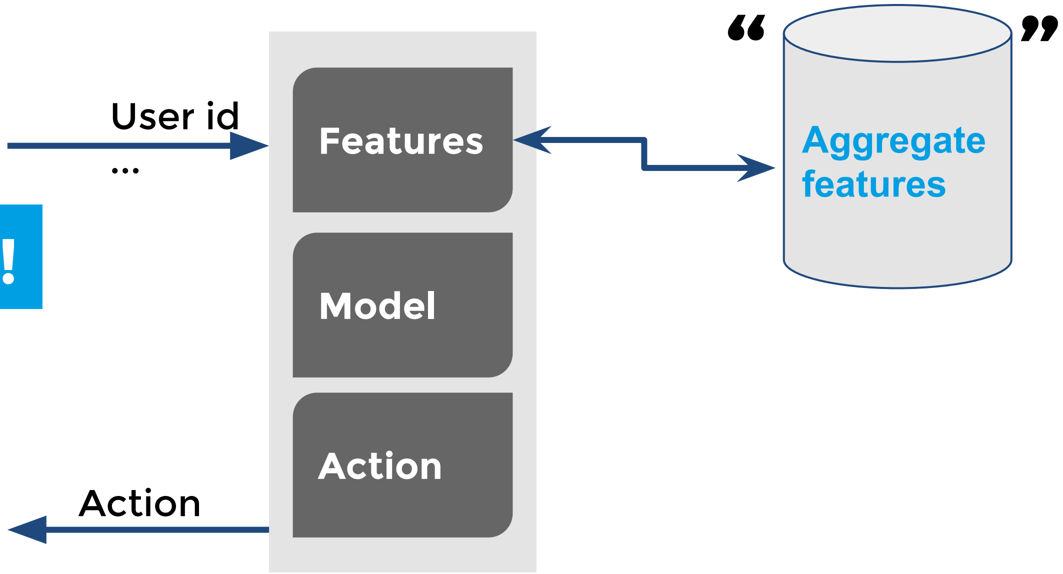
Behind the Scenes.





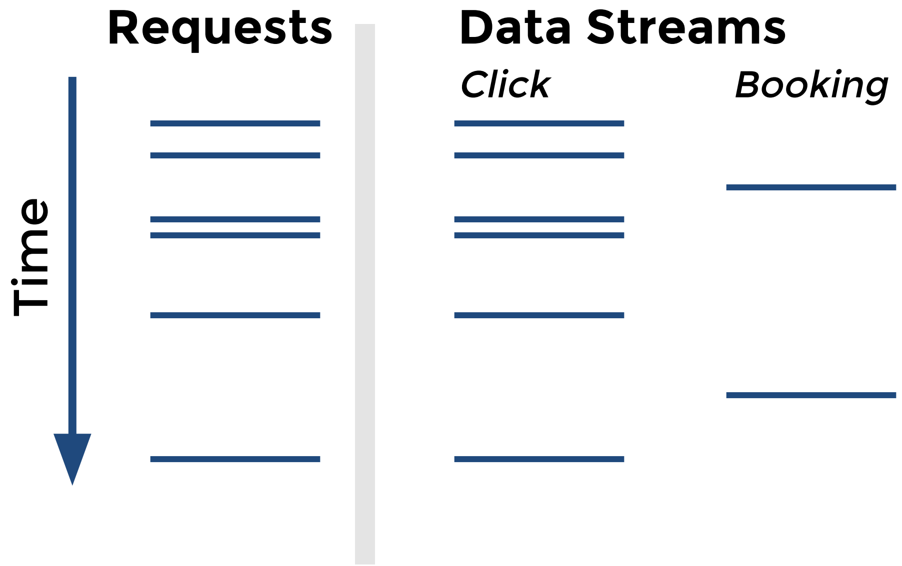
Behind the Scenes.



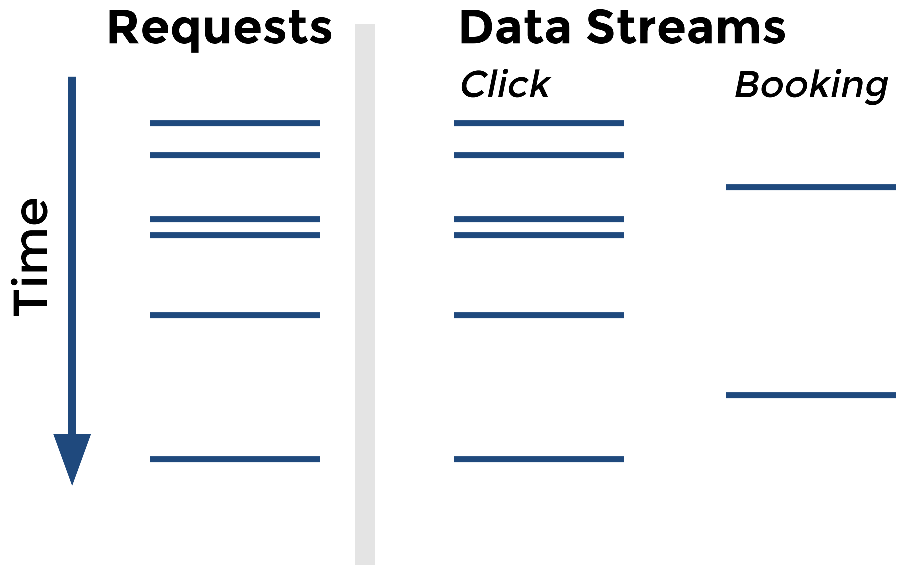


Aggregate Features

Aggregate Features.



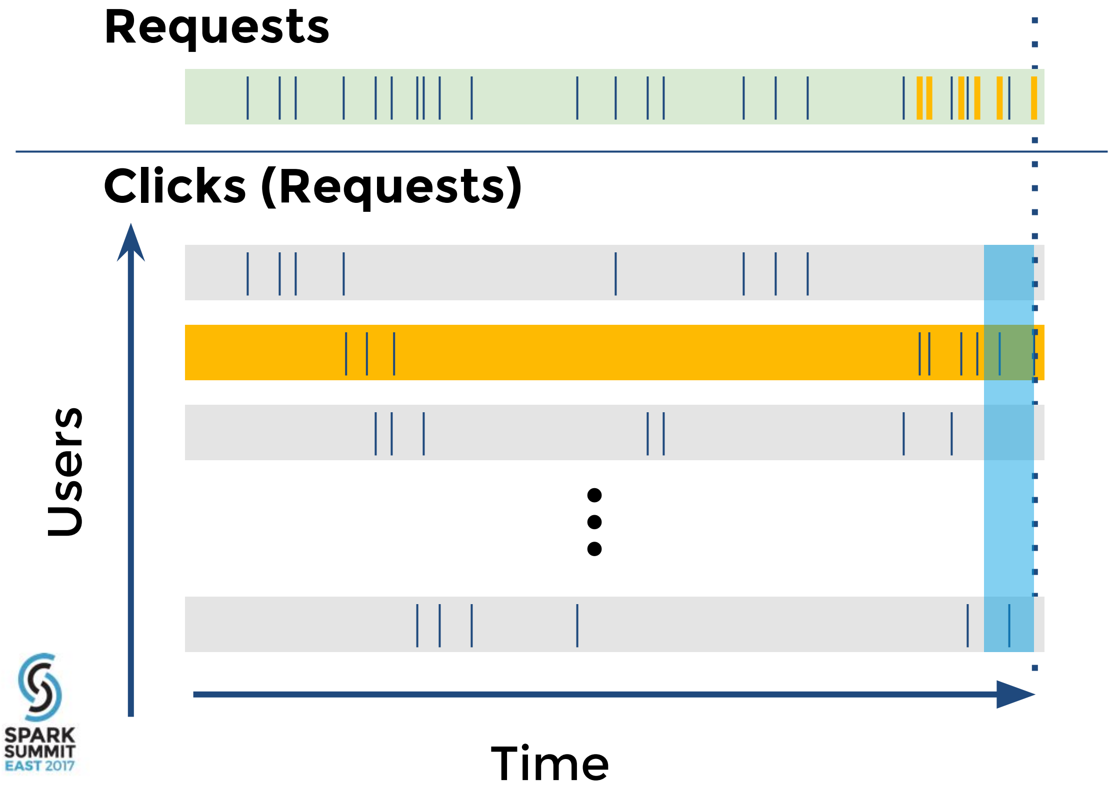
Aggregate Features.



Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - …
- Hotel Page
  - % booking per page view last 3 months
  - …

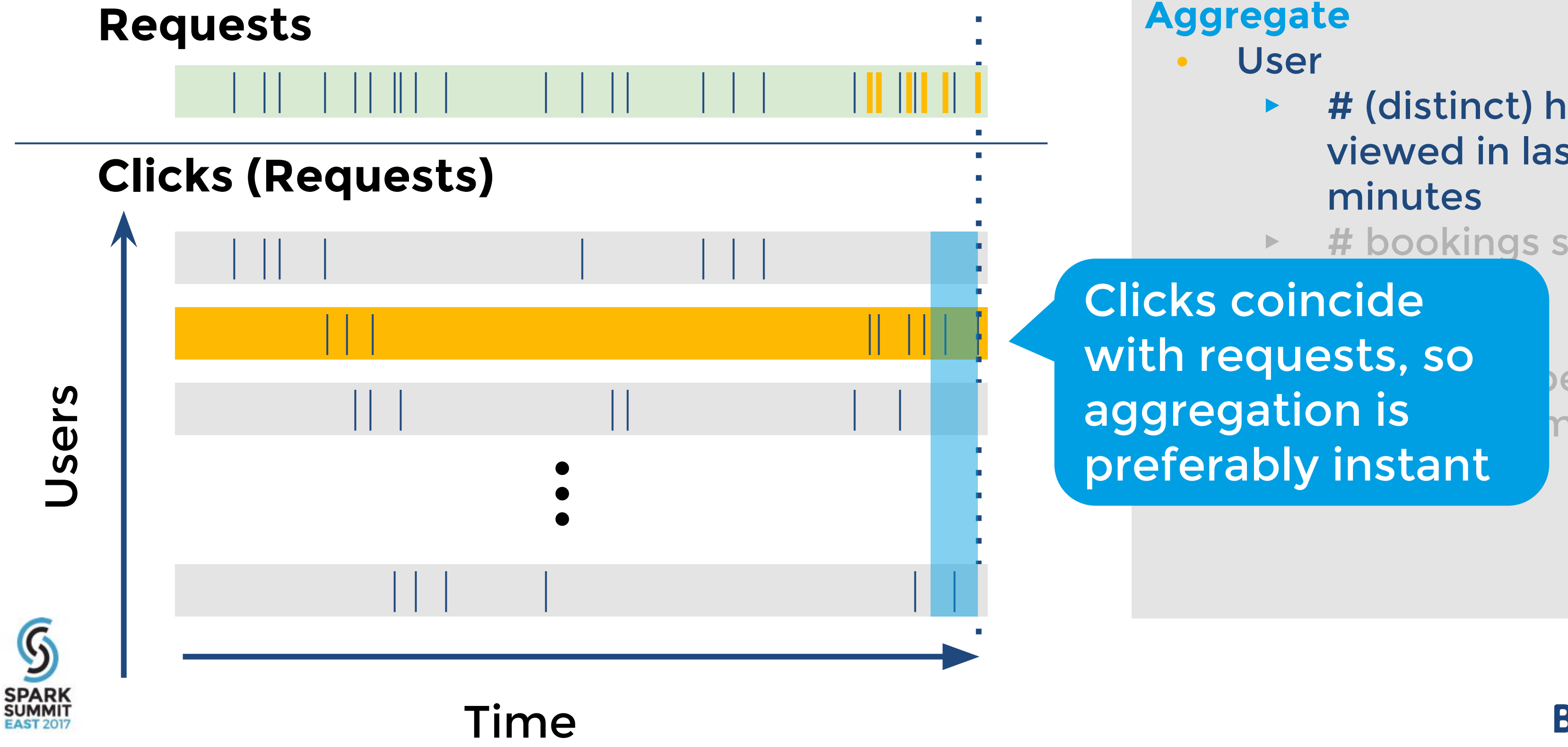
Aggregate Features.



Aggregate

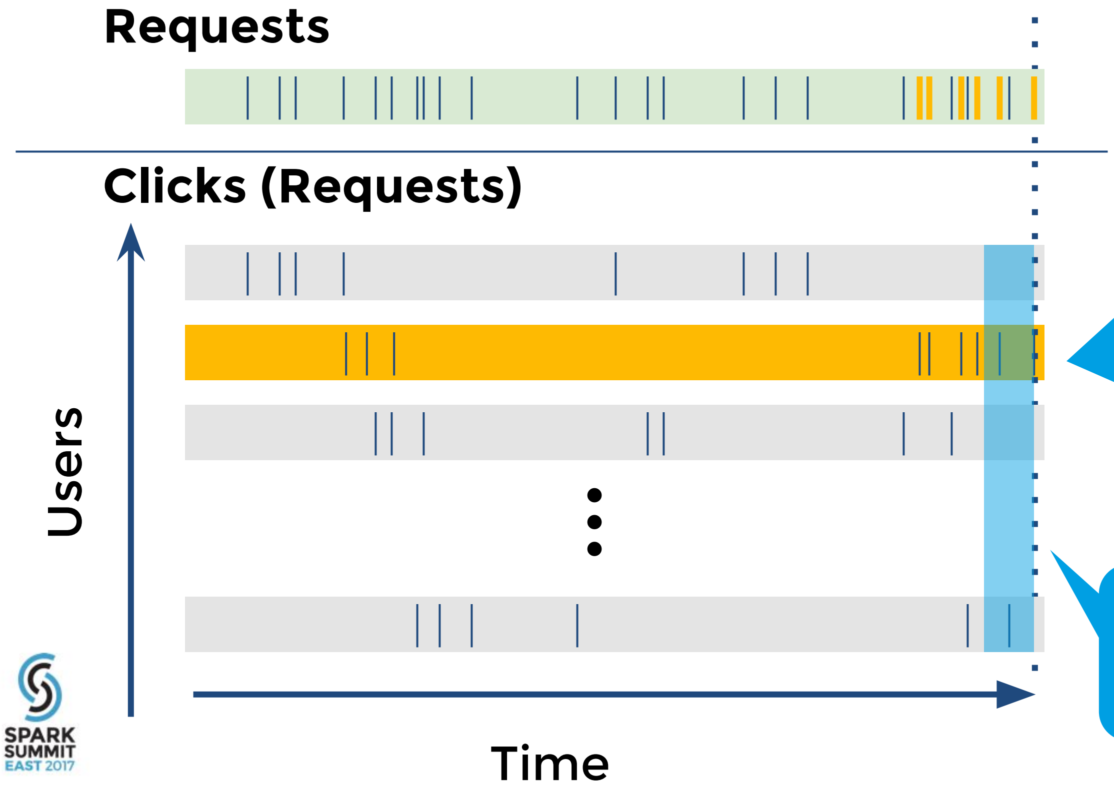
- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

Aggregate Features.



Aggregate
- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...per page
  - ...months

Aggregate Features.



Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far

per page
months

Clicks coincide with requests, so aggregation is preferably instant

Small time window, so limited amount of data

Aggregate Features.

EsperTech

```
SELECT
    COUNT(DISTINCT hotel_id)
FROM cliks.win:time(30 min)
GROUP BY user_id
```

In memory Complex Event Processing

- No lag: instant aggregation
- Scalability: see Esper website
- Not persistent

Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

Aggregate Features.

EsperTech

```
SELECT
    COUNT(DISTINCT hotel_id)
FROM cliks.win:time(30 min)
GROUP BY user_id
```

In memory Complex Event Processing

Aggregate

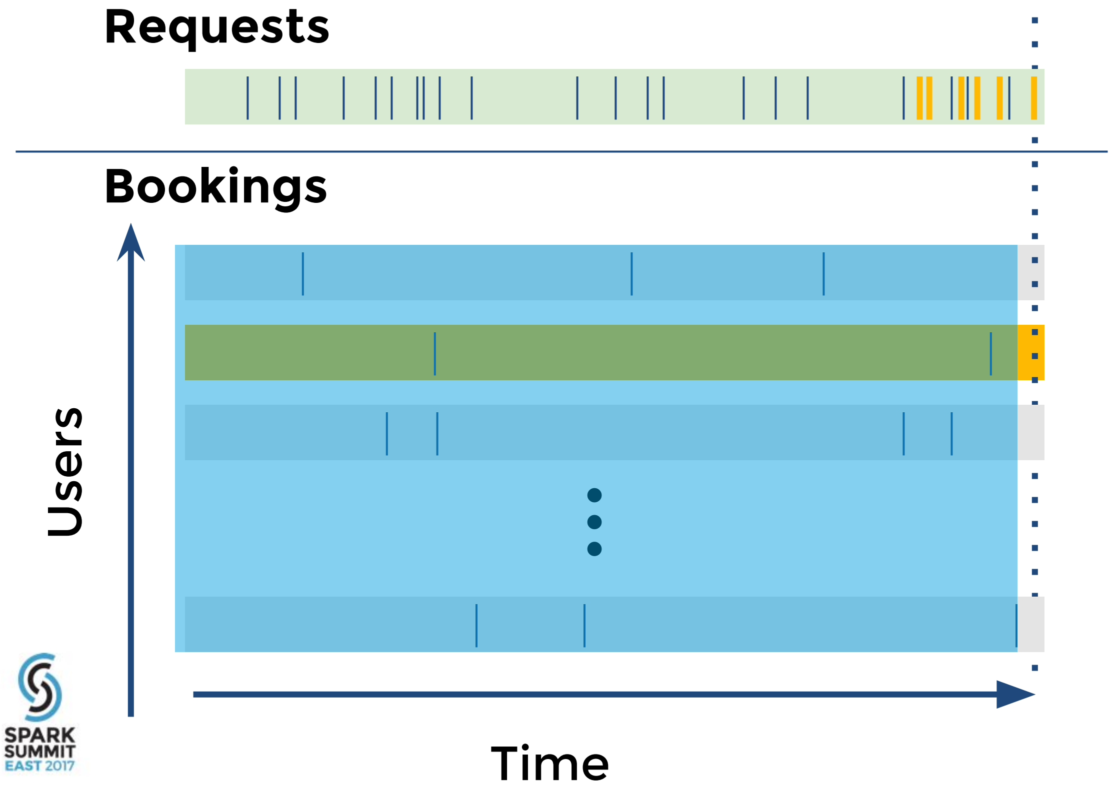
- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - …
- Hotel Page
  - % booking per page view last 3 months

From http://espertech.com/esper/faq_esper.php#streamprocessing

"Complex Event Processing and Esper are standing queries and latency to the answer is usually below 10us with more than 99% predictability."

"The first component of scaling is the throughput that can be achieved running single-threaded. For Esper we think this number is very high and likely between 10k to 200k events per second."

Aggregate Features.



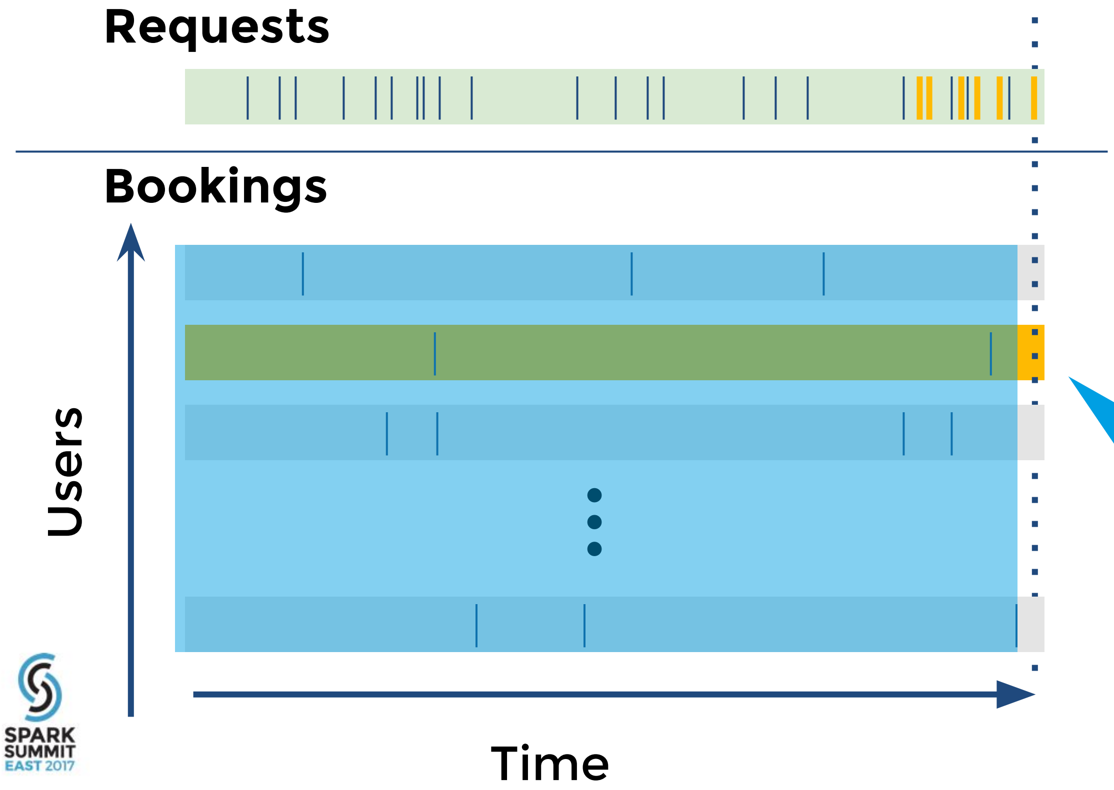
Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

SPARK SUMMIT EAST 2017

Booking.com

Aggregate Features.



Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - …
- Hotel Page

Very High Cardinality: features for every user who made at least one booking

Aggregate Features

High Cardinality Features with Cassandra

- Very scalable: reads & writes
- None-instant aggregations
  - Consistency fundamentally bound by gossip protocol
- Persistent

Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

Aggregate Features

(kafka → Spark Streaming) → cassandra

High Cardinality Features with Cassandra

From http://cassandra.apache.org/

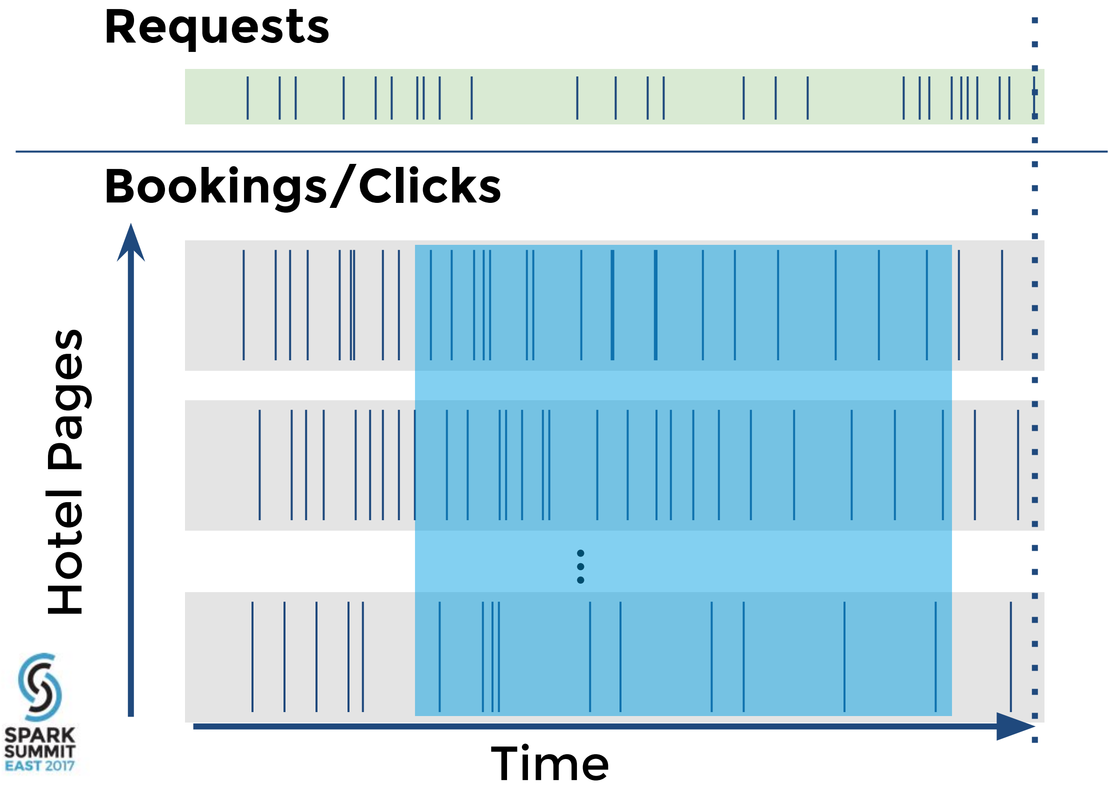
- Proven
- Fault Tolerant
- Performant
- Scalable
- Elastic
- ...

“Some of the largest production deployments include Apple's, with over 75,000 nodes storing over 10 PB of data, Netflix (2,500 nodes, 420 TB, over 1 trillion requests per day), …”

Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

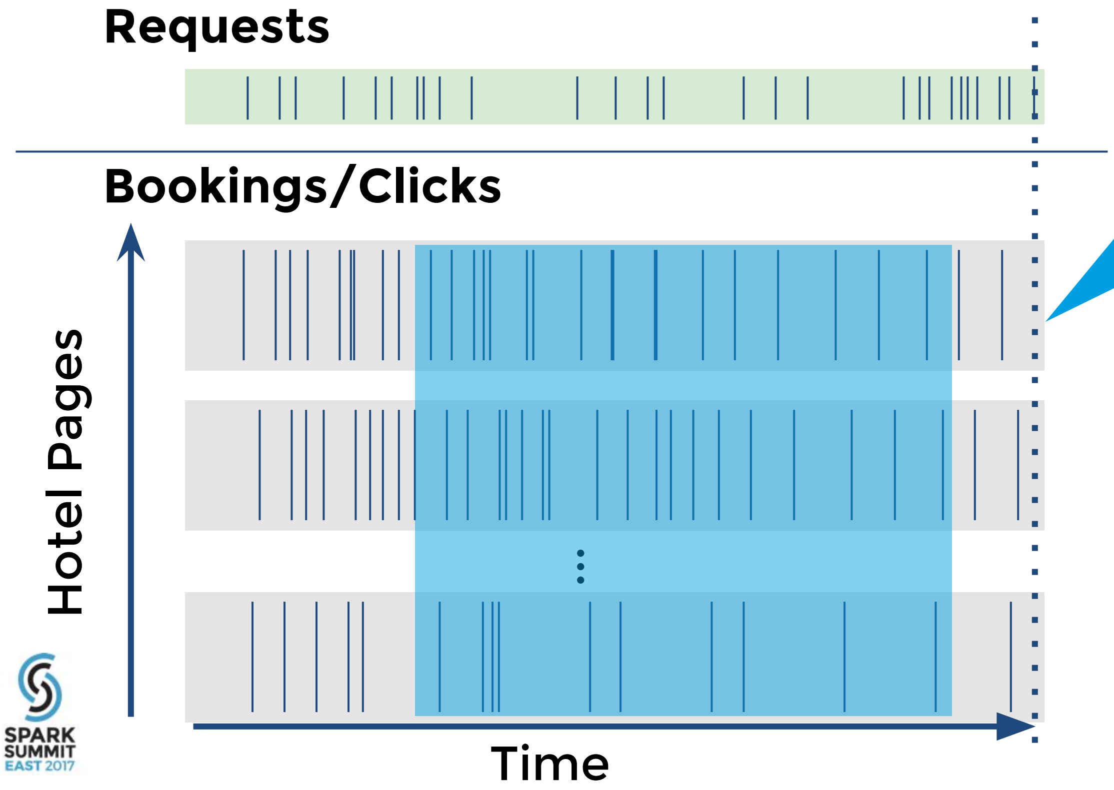
Aggregate Features.



Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

Aggregate Features.



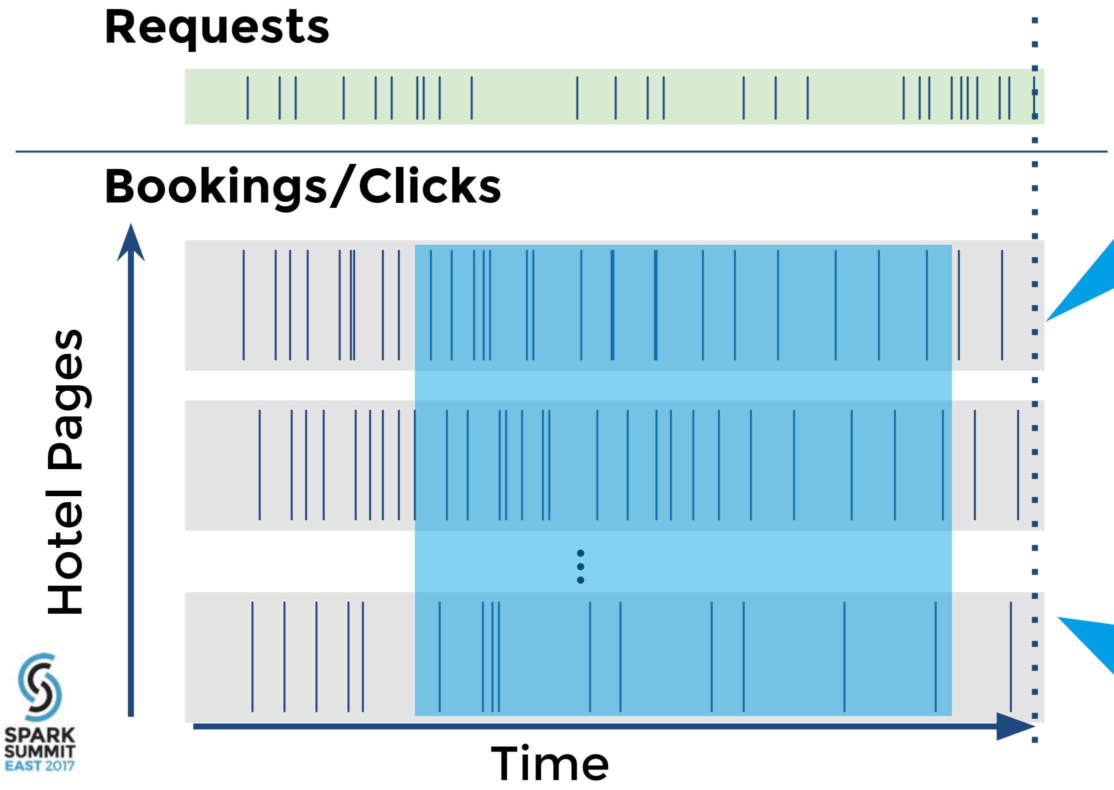
Aggregate

- User

Medium Cardinality: limited by number of hotels

- Hotel Page
  - % booking per page view last 3 months
  - ...

Aggregate Features.



Aggregate

- User

Medium Cardinality: limited by number of hotels

- Hotel Page
  - % booking per page view last 3 months

Larger lag is ok with time window of 3 months and large number of events per page.

Aggregate Features

```
SELECT
    page_id
    , COUNT(*) AS res_count
FROM reservations
GROUP BY page_id
```

Low to Medium Cardinality

- No need to go "fancy"
- Batch processing

MySQL

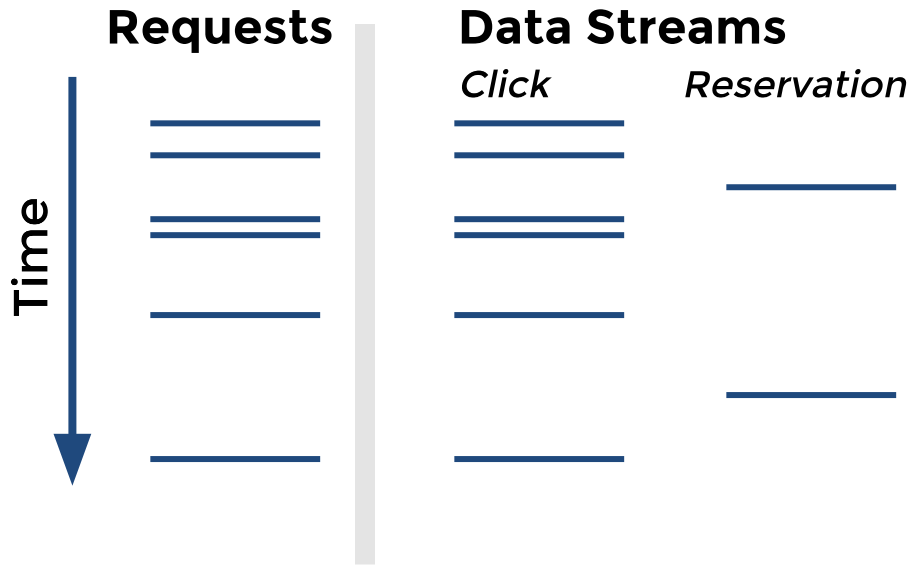
Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - …
- Hotel Page
  - % booking per page view last 3 months
  - …

Models

Training and deployment

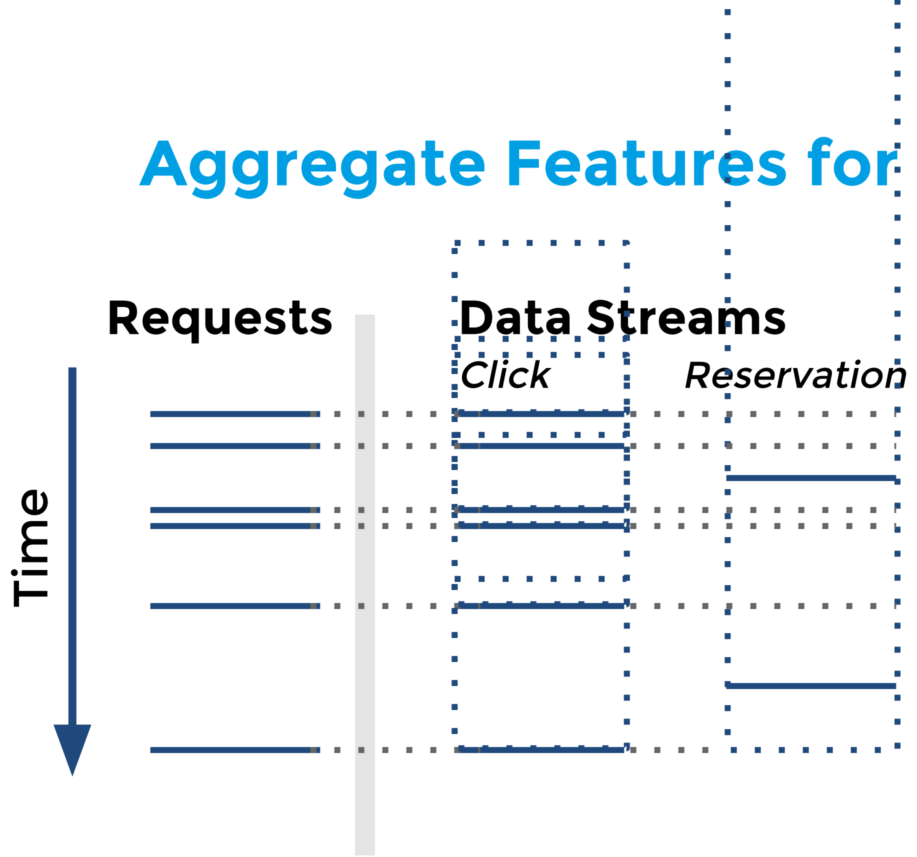
Aggregate Features for Training.



Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - …
- Hotel Page
  - % booking per page view last 3 months
  - …

Aggregate Features for Training.



Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - …
- Hotel Page
  - % booking per page view last 3 months
  - …

Aggregate Features for Training.

```
SELECT
     request.request_id
     , COUNT DISTINCT event.page_id
FROM request
JOIN event ON
     request.user_id = event.user_id
WHERE event.epoch BETWEEN
     request.epoch
     AND request.epoch + 30*60
GROUP BY request.request_id
```

Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

SPARK SUMMIT EAST 2017

Booking.com

Aggregate Features for Training.

```
SELECT
    request.request_id
    , COUNT DISTINCT event.page_id
FROM request
JOIN event ON
    request.user_id = event.user_id
WHERE event.epoch BETWEEN
    request.epoch
    AND request.epoch + 30*60
GROUP BY request.request_id
```

Warning: stragglers ahead! Distribute wisely!

Aggregate

- User
  - # (distinct) hotels viewed in last 30 minutes
  - # bookings so far
  - ...
- Hotel Page
  - % booking per page view last 3 months
  - ...

Aggregate Features for Training.

```
SELECT
    request.request_id
    , COUNT DISTINCT event.page_id
FROM request
JOIN event ON
    request.user_id = event.user_id
WHERE event.epoch BETWEEN
    request.epoch
    AND request.epoch + 30*60
GROUP BY request.request_id
```

Scalable Technologies

- Simple SQL
  - Hive
- Facing **stragglers**
  - Spark

Model Training & Iteration

Train Model

Improve Model

y X

- Features
  - Engineering
  - Selection
- Hyper parameters

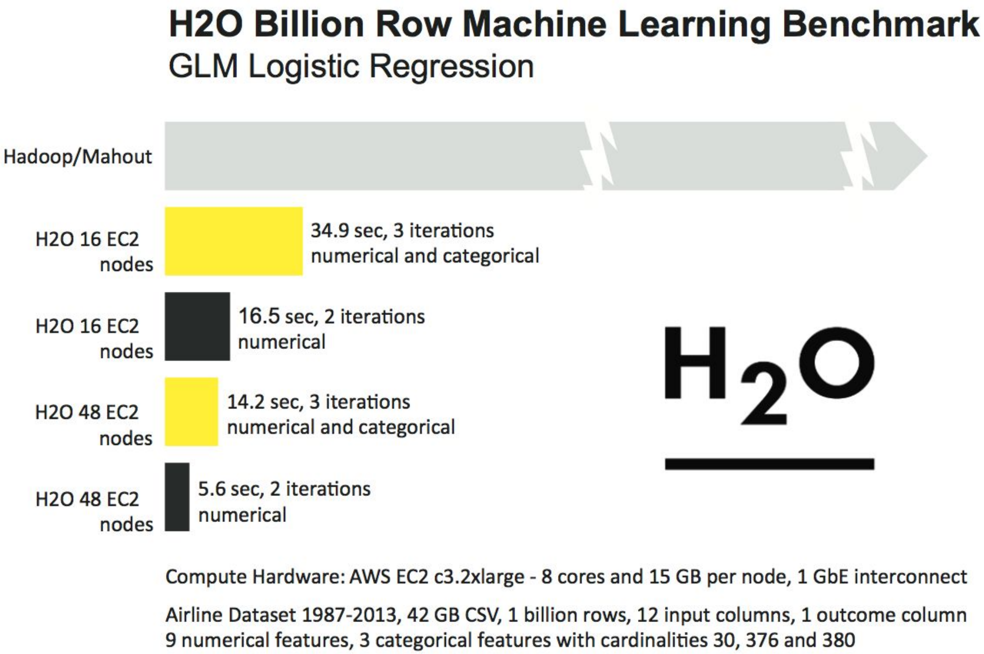
Spark + H_2O
SPARKLING WATER

Evaluate

	Predict YES	Predict NO
Actual YES	TP	FN
Actual NO	FP	TN

Investigate

Sorry: H2O?



The benchmark shown on this slide can be found in:
http://h2o-release.s3.amazonaws.com/h2o/rel-lambert/5/docs-website/resources/h2odatasheet.html
More benchmarks:
https://github.com/szilard/benchm-ml

Time to Deploy

Spark + H_2O

SPARKLING WATER

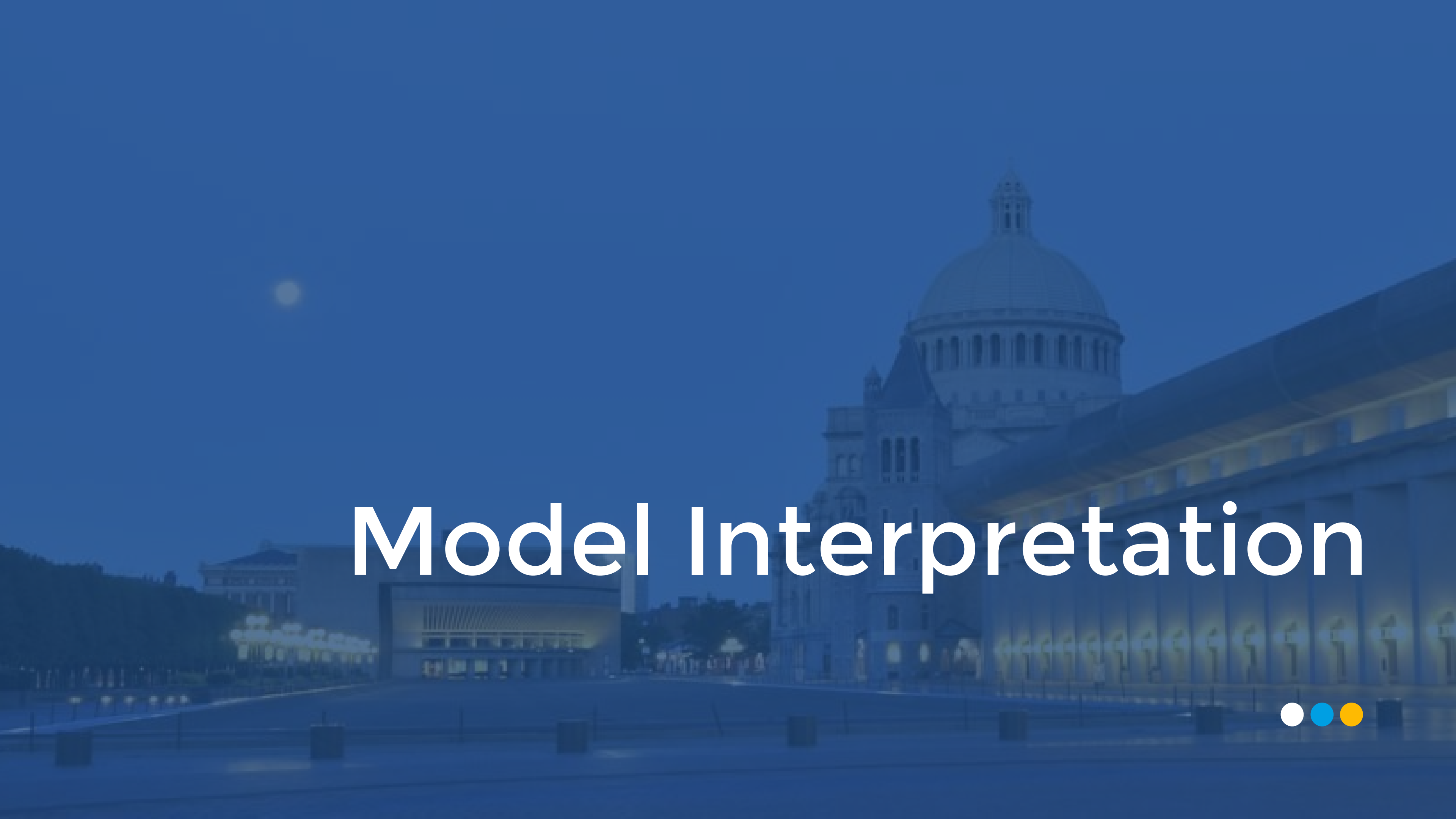
```
model = H2ORandomForestEstimator(ntrees=50, max_depth=10)
model.train(x=inputCols, y="label",
            training_frame=trainingDf,
            validation_frame=validationDf)
model.download_pojo("/my/model/path/")
```

Time to Deploy.

```
model = H2ORandomForestEstimator(ntrees=50, max_depth=10)
model.train(x=inputCols, y="label",
            training_frame=trainingDf,
            validation_frame=validationDf)
model.download_pojo("/my/model/path/")
```

AWESOME!!!

POJO: Plain Old Java Object. Model export in plain Java code for real- time scoring on the JVM. Very fast!



Model Interpretation

Score Interpretation.

Logistic Regression

$\beta x = -5.12 \times$ feature 1 $+ 13.9 \times$ feature 2 $+ \ldots$

Contribution Feature i: $\beta_i x_i$

Decision Trees

- feature 15 > 9 — p_0
  - yes: feature 2 > 0.56 — p_1
    - yes: p_3
    - no: p_4
  - no: feature 4 > 200 — p_2
    - yes: p_5
    - no: feature 11 > 6 — p_6
      - yes: p_7
      - no: p_8

Contribution Feature i: $p_{node} - p_{parent\ node}$

Example:
$p_3 = p_0 + (p_1-p_0) + (p_3-p_1)$
p_0: bias
(p_1-p_0): feature 15 contr.
(p_3-p_1): feature 2 contr.

Decision tree interpretation based on:
http://blog.datadive.net/interpreting-random-forests

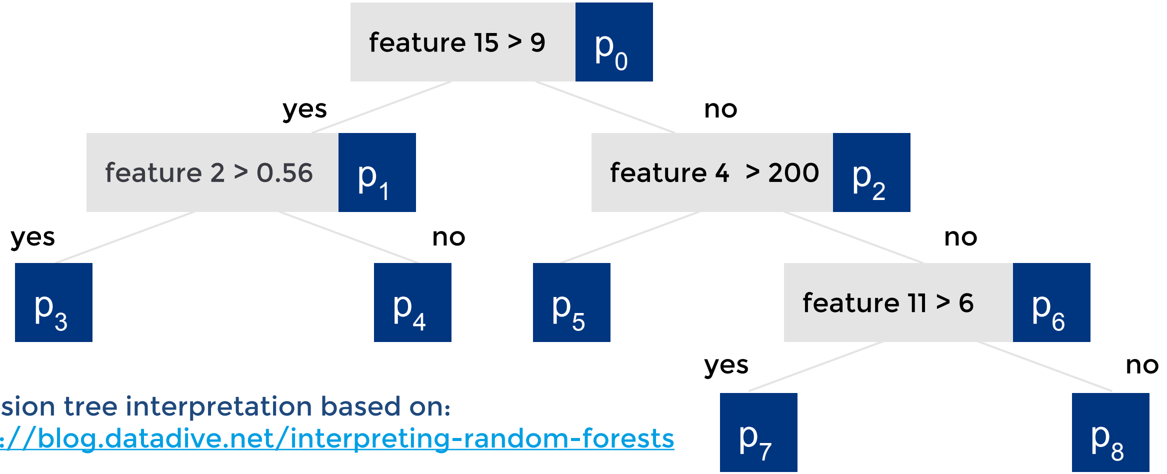
Score Interpretation.

Logistic Regression

βx = -5.12 x feature 1 + 13.9 x feature 2 + ...

Contribution Feature i:
$\beta_i x_i$

Decision Trees

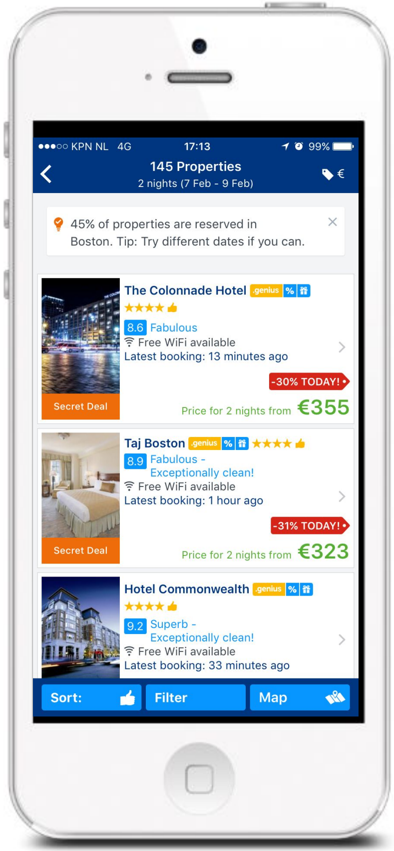


Contribution Feature i:
$p_{node} - p_{parent\ node}$

We hacked this interpretation into Spark ML during a Hackathon at Booking:) H2O does not offer it (yet?).

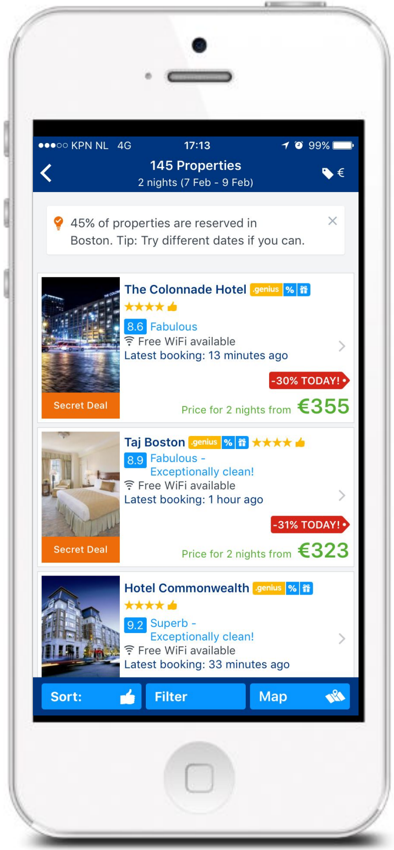
Decision tree interpretation based on:
http://blog.datadive.net/interpreting-random-forests

Probability to Book.



The probability of booking is high; largest contribution from #distinct property pages. Let's show a summary!

Probability to Book.

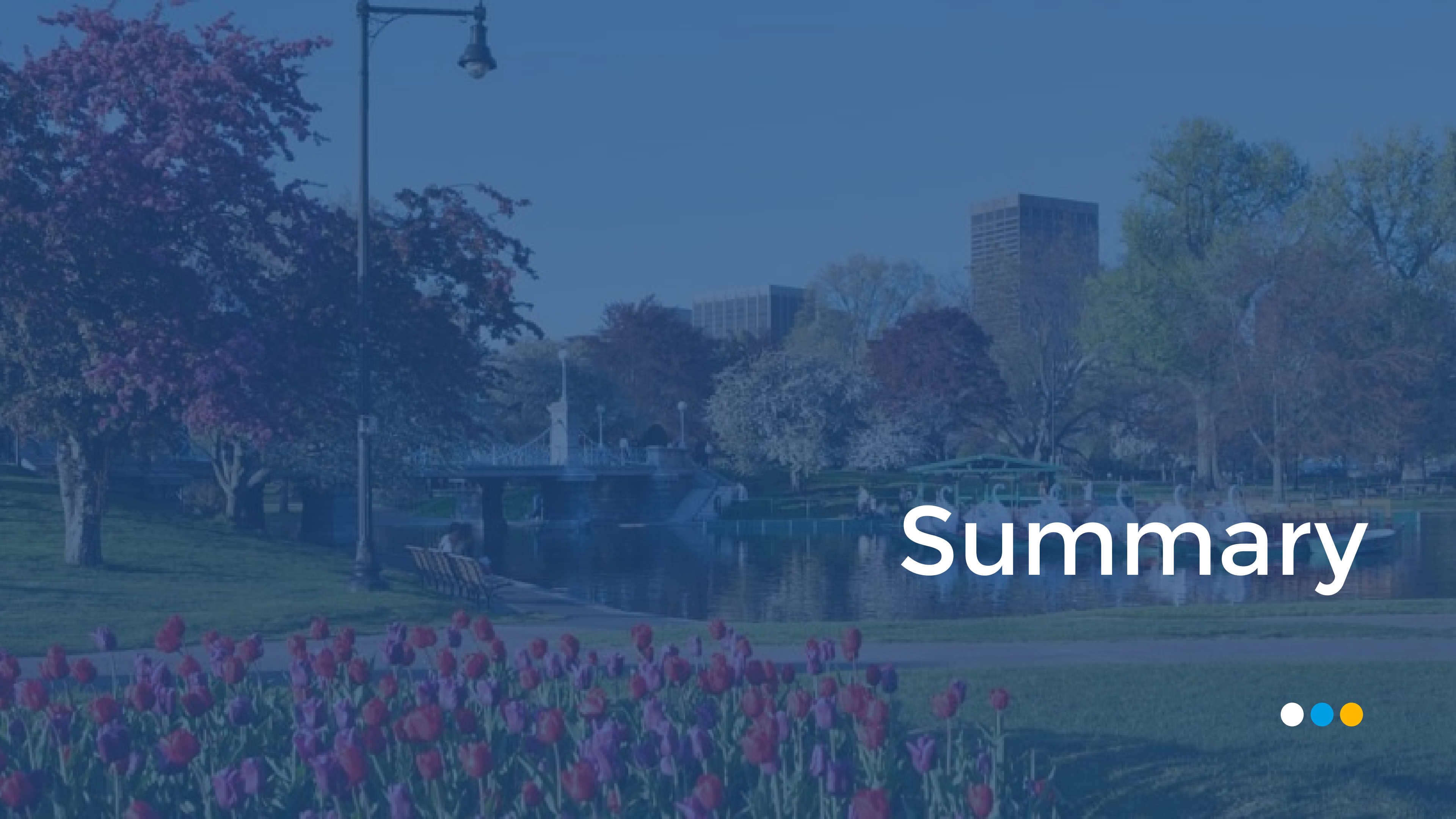


Awesome! This app really helps me book!

The probability of booking is high; largest contribution from #distinct property pages. Let’s show a summary!

SPARK SUMMIT EAST 2017

Booking.com



Summary

Summary.

- How to make Aggregate Features available?
  - Long/short time windows
  - High and Low cardinality dimensions
- Model Training and Deployment
  - H2O POJO for fast Real-Time scoring
- Interpretation
  - Contributions per Feature per Score

Thank You.

Questions?

We’re hiring!

Kees Jan de Vries from Booking.com

linkedin.com/in/kees-jan-de-vries-93767240