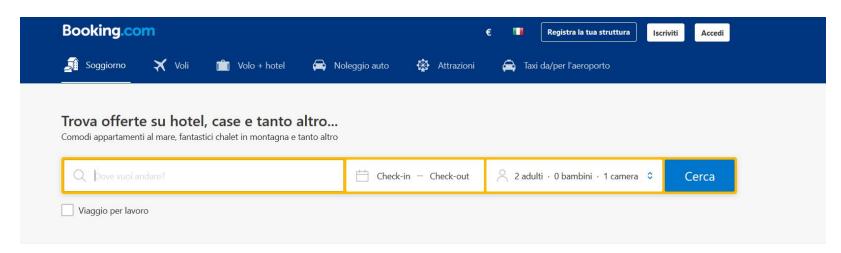
# Booking.com

Recommendation system

Fabrizio Mazzone - Philipp Schlieker

### The context

Booking.com allows the user to book hotels from everywhere in the world.



## The ranking

Typing a city, you can see the results basing on several ranking filters.

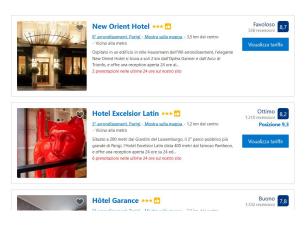
Our top picks	Show homes first	Price (lowest first)	Stars 🔻	Star rating and price	Distance from city centre	
	The state of the s		122			

### Top picks



We will focus on what Booking defines *our top picks*. This ranking is a recommendation system based on many factors:

- Room availability
- Preferred Partners
- Reviews
- Booking earning margin



### **Room availability**

Booking merges data from internal booking and manual updates to know about rooms availability in every hotel.

Of course, hotels that are full should not be recommended.



### Reviews

The reviews in Booking.com are in the range from 0 to 10, with a decimal digit.

They are usually sided by a very short description.







### **Preferred partners**

The preferred partner program is proposed to the user to provide guests a positive experience thanks to its commendable service and good value.

The hotel could pay Booking.com a bit more to be in that program.



### **Booking earning margin**

Booking.com should make money.

Well, Booking .com actually makes money through charging hotels for selling their inventories. It takes a little commission depending on the city, the hotel, the room.

Of course, it tries to push up the results where it can earn more.



# Users are important for booking.com

### User preferences

Users decide which hotel to book focusing on:

- Price Sensitivity
- Business vs. Leisure
- Accompaniment
- Rating of the hotel
- Location of the hotel



\_

## What we did

We want to simulate Booking.com recommendation system based on what we set in the previous slides.

Our aim is obtaining the lowest regret and thereby maximizing *booking.com* margin.



#### What we need

#### **→** Environment

All the elements that simulates our website (Booking.com)

#### → Some agents

An agent states the strategy used to suggest a hotel.

#### → Statistics

We did some comparison in order to know the best agent to choose for our specific environment

### Our environment

# RoomAvailability

Every hotel has a different room availability.

#### 2. Partner Hotels

Some hotels are featured and they should appear more than the others.

# 3. Earning Margins

If the user likes the hotel, the margin is different for every hotel.

# Some agents

#### 1. Random

It performs a completely **random** choice, respecting the constraints, though.

#### 2. Adaptive Greedy

At first it randomly chooses, then it takes the hotel that had the **average** greatest reward.

#### 3. Epsilon Greedy

With a **eps** probability, it picks randomly, than exploits the learned **feature** to choose the best hotel.

#### \_

# Some agents

# 4. Thompson Sampling

It calculates and updates the estimated reward based on **probability** distribution.

# 5. Upper Confidence Bound

It samples very quickly to reduce the **uncertainty** of the unknowns before **locking** onto the most profitable hotel..

# Some agents

# 6. Embedding Agent Offline

It trains embeddings based on **historical data** and learns expected payouts.

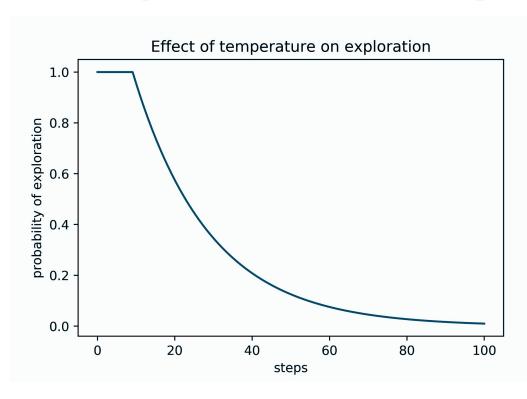
# 7. EmbeddingAgent Offline /Online

Embedding model that retrains when 30% new data is available.

# 8. Epsilon Embedding Agent

It performs a **weighted pick** between random exploration and exploitation of learned embeddings.

### Effect of Temperature on Exploration



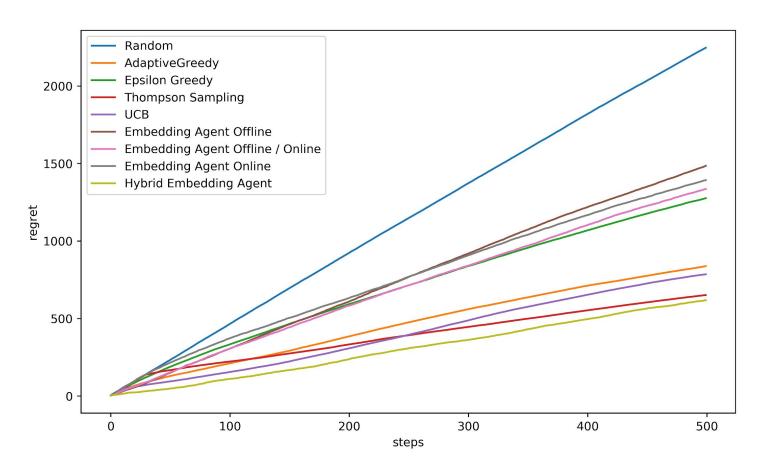
\_

# Some agents

#### 9. Hybrid Agent

Uses Embeddings but also takes Meta-Data into account

#### **Our results**



### **Further improvements**

- Implement Linear Stochastic bandits (LinUCB, LinTS)
- Perform Grid Search on Parameter Space
- Experiment with different ways to reduce temperature