Master on Foundations of Data Science

Recommender Systems

Context Based Recommendations

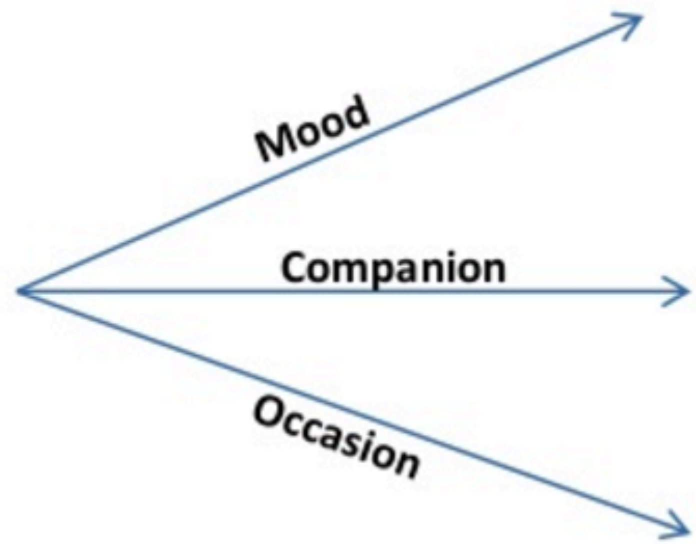
Santi Seguí | 2018-2019


CRISPY

Any movie you prefer to see at **home** than at the **theater**?

and viceversa?

Non-Context vs. Context



Decision = Rational + Contextual

Context Based Recommenders

- Sensitive to additional information that defines the specific situation under which recommendations are made:
  - **Time:** weekends, holidays, night/day, summer/winter
    - Recommendation that is relevant to the morning context, may not be relevant in the evening and vice-versa.
  - **Location:** A traveling user might wish to determine a recommendation for a restaurant in her location
  - **Social information:** Social context is important. For example, the choice of a user's friends, tags, and social circles can affect the recommendation process

Context Based Recommenders

- Examples:
  - **Travel destination**: in winter vs. summer
  - **Movie watching**: with children vs. with partner
  - **Restaurant**: quick lunch vs. business dinner
  - **Music**: workout vs. study

What is context?

- "**Context** is any information that can be used to characterize the situation of an entity" by Anind K.Dey, 2001

How Contextual Factors Change	Knowledge of the RS about the Contextual Factors		
	Fully Observable	Partially Observable	Unobservable
Static	Everything Known about Context	Partial and Static Context Knowledge	Latent Knowledge of Context
Dynamic	Context Relevance Is Dynamic	Partial and Dynamic Context Knowledge	Nothing Is Known about Context

- Representative context: Fully observable and static
- Interactive Context: Non-fully observable and Dynamic

Context Dimension

- Explicit:
  - Location, Time, Device, Language,…
- Implicit:
  - Mood, Companion,…

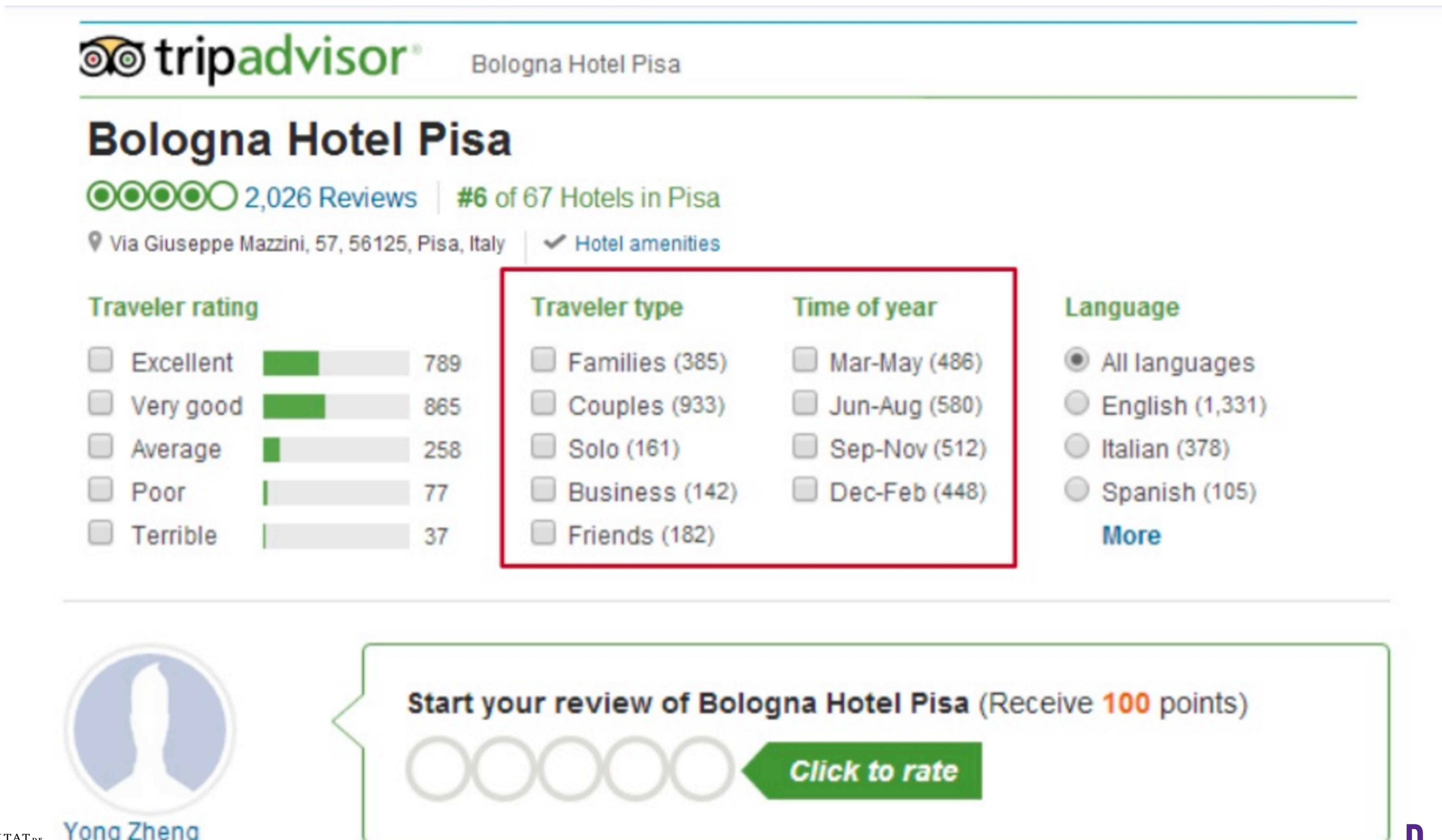
“**Contextual factors** can greatly influence the utility of recommendations for users. In many recommendation and personalization applications, particularly in domains where user context changes dynamically, it is difficult to represent and **model contextual** factors directly, but it is often **possible** to observe their impact on user preferences **during the course of users’ interactions with the system**”

Context adaptation in interactive recommender systems
N Hariri, B Mobasher, R Burke
Proceedings of the 8th ACM Conference on Recommender systems, 41-48
18 2014

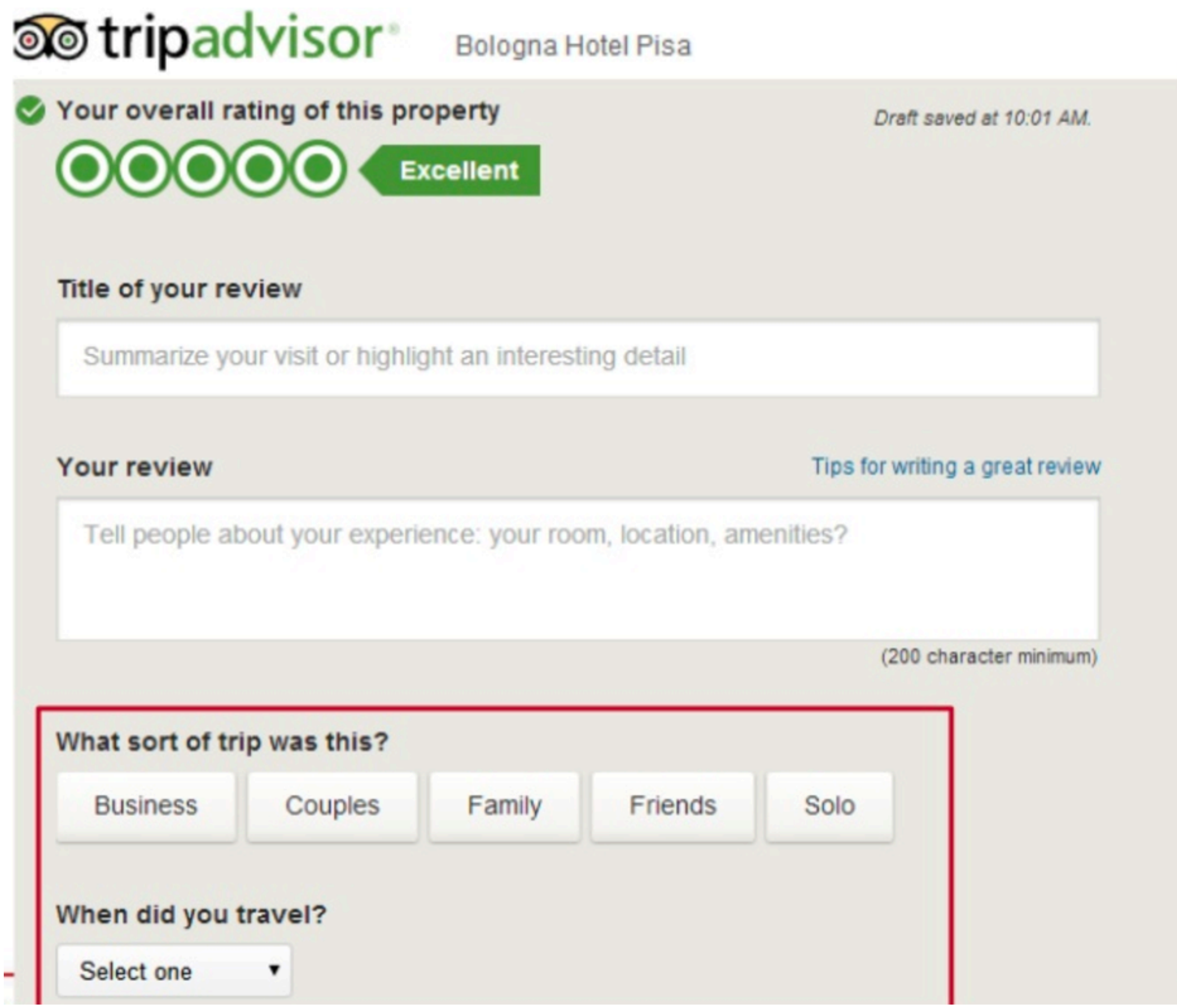
UNIVERSITAT DE BARCELONA

Context Acquisition (Real Time)

Context Acquisition (Explicit)



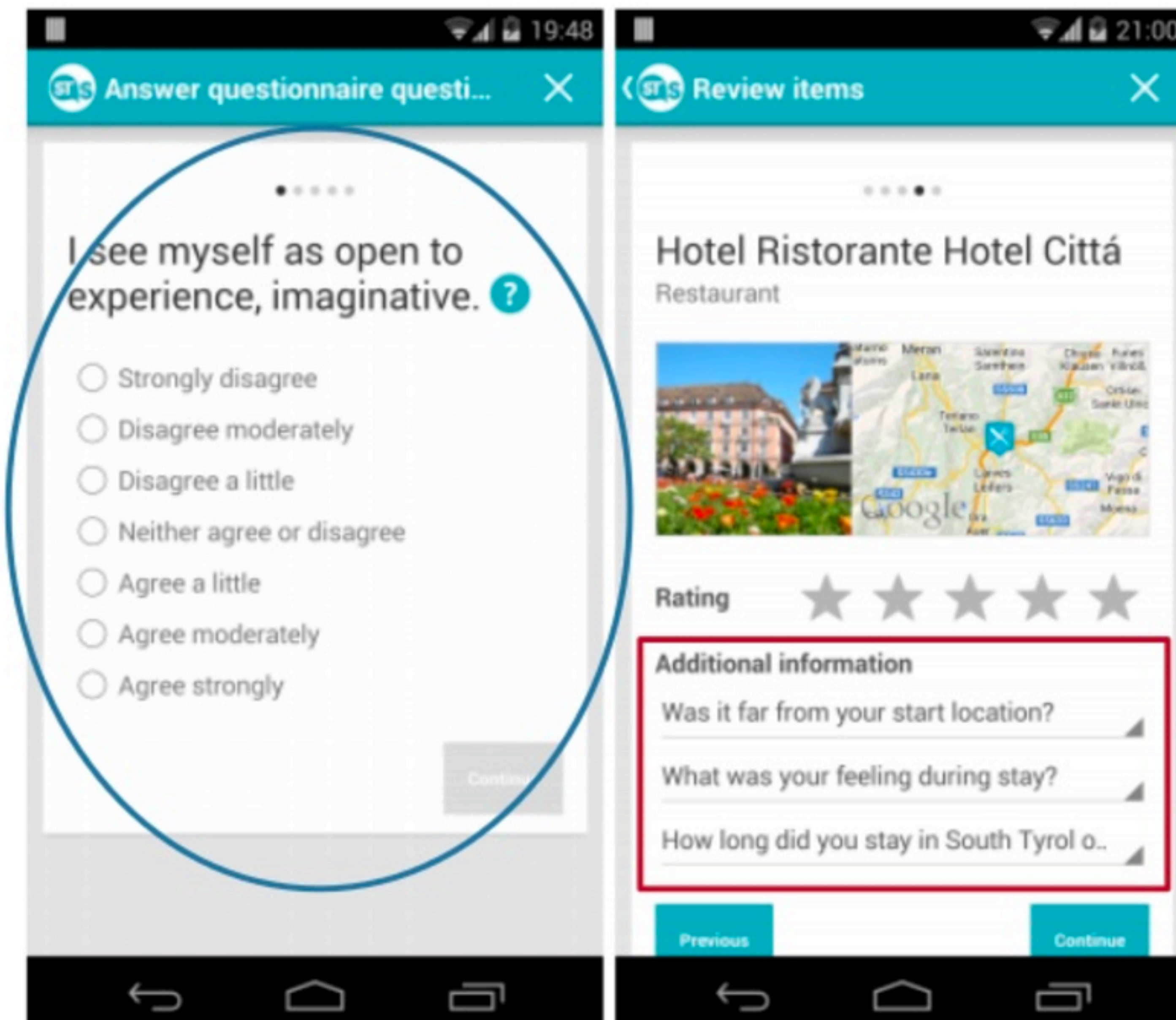
Context Acquisition (Explicit)



Context Acquisition (Explicit)

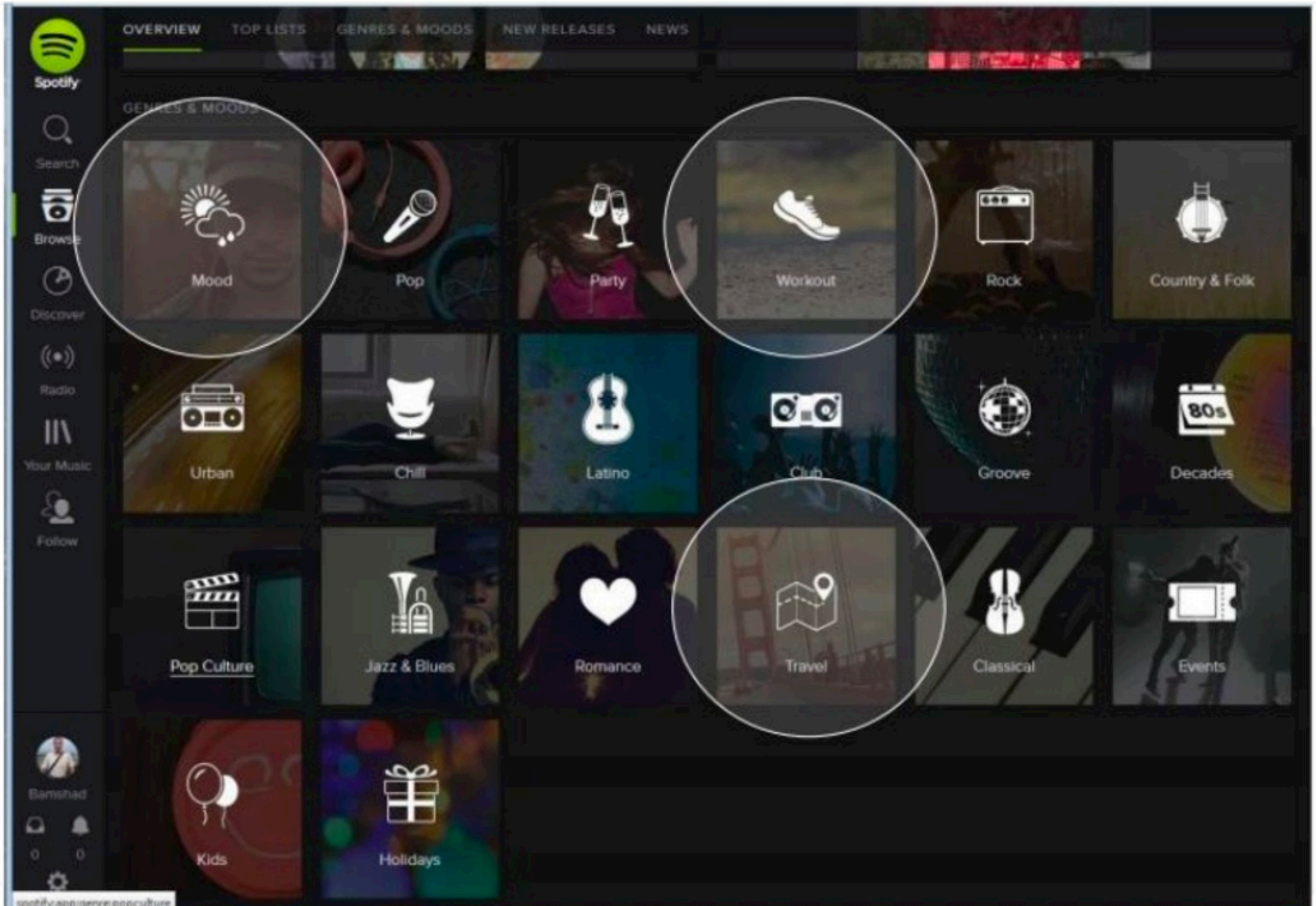
Mobile App: South Tyrol Suggests

Personality Collection



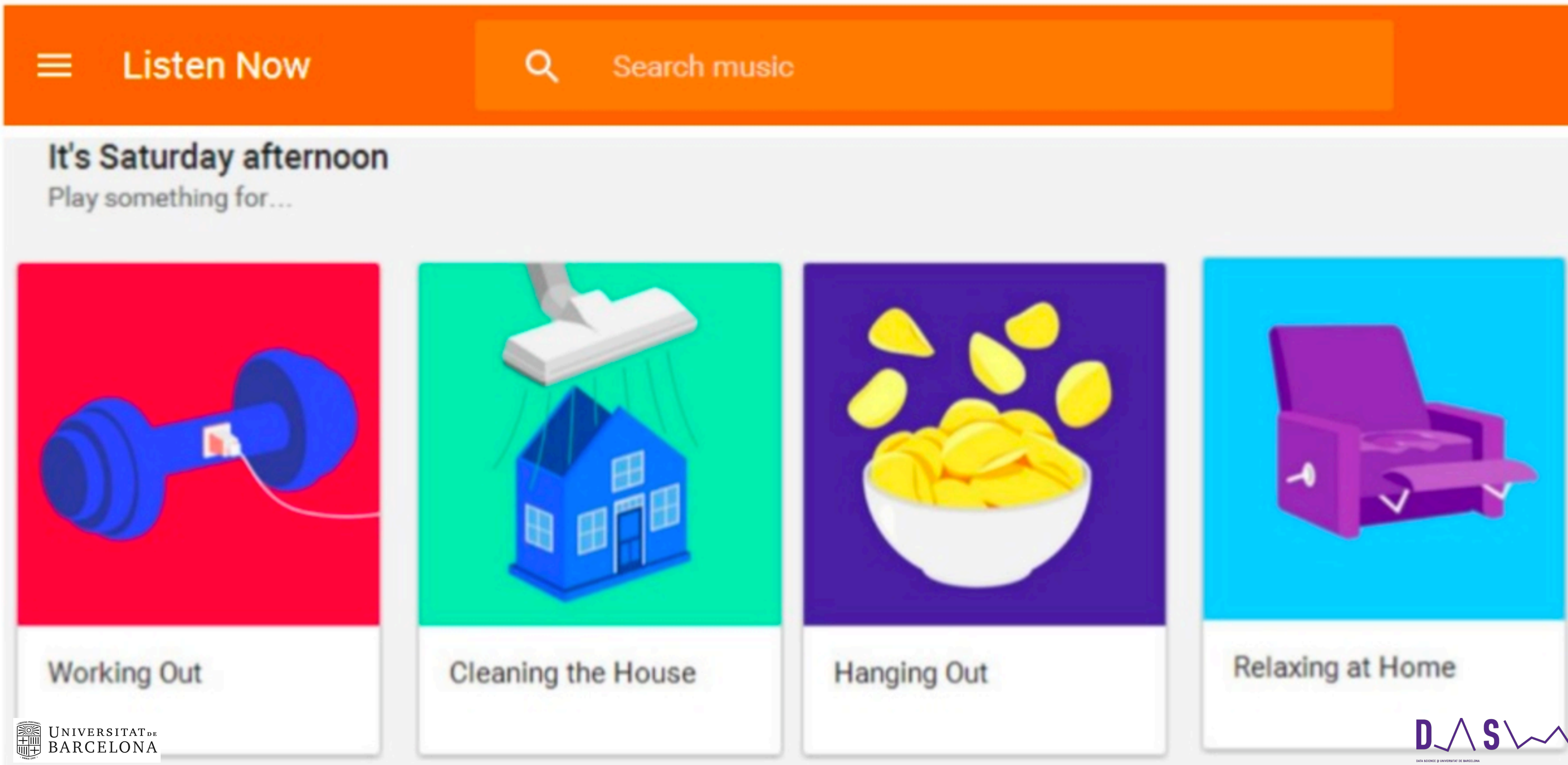
Context Collection

Context Acquisition (Predefined)



Context Acquisition (Predefined)

Google Music: Listen Now



Context Acquisition

- How to **collect the context** and user preferences in contexts?
  - **By users surveys or explicitly asking for user inputs**
  - **By usage data**
    - The log data usually contains time and locations (at least);
    - User behaviors can also infer context signals;
    - Cross domain applications

Which context is important?

- Not all of the context are relevant or influential. How can we find the importance of the context?

Contextual information that does not have a significant contribution explaining the variance in users' decisions could degrade the prediction, since it could play the role of noise

- By Statistical Analysis or Detection on Contextual Ratings
    - *Statistical test, Mutual Information, Information Gain*

Context for **Movie** Recommender System

Which is the important contextual information?

How can we estimate it?

Relevant context in a movie recommender system: Users' opinion vs. statistical detection
A Odic, M Tkalcic, JF Tasic, A Košir
ACM RecSys 12 | 40 | 2012

Universitat de Barcelona

Context for
Movie Recommender System

Context for
Trivago Recommender System

Context for **Movie** Recommender System

Contextual variable	Description
time	morning, afternoon, evening, night
daytype	working day, weekend, holiday
season	spring, summer, autumn, winter
location	home, public place, friend's house
weather	sunny/clear, rainy, stormy, snowy, cloudy
social	alone, partner, friends, colleagues, parents, public, family
endEmo	sad, happy, scared, surprised, angry, disgusted, neutral
dominantEmo	sad, happy, scared, surprised, angry, disgusted, neutral
mood	positive, neutral, negative
physical	healthy, ill
decision	user picked the item, item suggested by other
interaction	first, n-th

Relevant context in a movie recommender system: Users' opinion vs. statistical detection
A Odic, M Tkalcic, JF Tasic, A Košir
ACM RecSys 12

Context for **Movie** Recommender System

how do we obtain the data?

- via user-survey
- via the detection from the rating data

Relevant context in a movie recommender system: Users' opinion vs. statistical detection A Odic, M Tkalcic, JF Tasic, A Košir ACM RecSys 12	40	2012

Context for Movie Recommender System

Table 3: Assessment and detection results.

Assessment		Detection	
Relevant	**Irrelevant**	**Relevant**	**Irrelevant**
time		daytype	
location	day type	location	
social	season	endEmo	time
endEmo	weather	domEmo	season
domEmo	physical	mood	weather
mood	decision	physical	social
interaction		decision	
		interaction	

Relevant context in a movie recommender system: Users' opinion vs. statistical detection
A Odic, M Tkalcic, JF Tasic, A Košir
ACM RecSys 12
40 2012

UNIVERSITAT DE BARCELONA

How to make a Context Aware Recommender Systems?

Context-aware RecSys

- **Traditional RS**: Users x Items -> Ratings
- **Contextual RS**: Uxers x Items x Contexts -> Ratings

User	Item	Rating	Time	Location	Companion
U1	T1	3	Weekend	Home	Kids
U1	T2	5	Weekday	Home	Partner
U2	T2	2	Weekend	Cinema	Partner
U2	T3	3	Weekday	Cinema	Family
U1	T3	?	Weekend	Cinema	Kids

Example of Multi-dimensional Context-aware Data set

Items at different context

At Cinema

At Home

At Swimming Pool

Item Splitting

User	Item	Location	Rating
U1	M1	Pool	5
U2	M1	Pool	5
U3	M1	Pool	5
U1	M1	Home	2
U4	M1	Home	3
U2	M1	Cinema	2

High Rating

Low Rating

Significant difference?
Let's split it !!!

Same movie, different IDs.

M1

M11: being seen at Pool

M12: being seen at Home

Item Splitting

User	Item	Loc	Rating
U1	M1	Pool	5
U2	M1	Pool	5
U3	M1	Pool	5
U1	M1	Home	2
U4	M1	Home	3
U2	M1	Cinema	2

Transformation

User	Item	Rating
U1	M11	5
U2	M11	5
U3	M11	5
U1	M12	2
U4	M12	3
U2	M12	2

Question:
How to find such a split? Pool and Non-pool, or Home and Non-home?
Which one is the best or optimal split?

Item Splitting

- **Splitting Criteria** (Impurity Criteria)
  - Impurity criteria and significance test are used to make the selection
  - Several measures:
    - t_mean(t-test), t_prop(z-test), t_chi (chi-square test), t_ig(Information gain)

User Splitting and UI Splitting

- Similarly, the splitting approach can be applied to users too!

    - **User Splitting:** Instead of splitting items, it may be useful to consider one users as two different users, if user demonstrates different preferences across context.

    - **UI Splitting:** simply a combination of item splitting and user splitting - both approaches are applied to create a new rating matrix - new users and new items are created in the rating matrix.

Splitting and Transformation

User	Item	Rating	Time	Location	Companion
U1	T1	3	Weekend	Home	Friend
U1	T1	5	Weekend	Cinema	Girlfriend
U1	T1	?	Weekday	Home	Family

(a) by Item Splitting

User	Item	Rating
U1	T11	3
U1	T12	5
U1	T11	?

(b) by User Splitting

User	Item	Rating
U12	T1	3
U12	T1	5
U11	T1	?

(c) by UI Splitting

User	Item	Rating
U12	T11	3
U12	T12	5
U11	T11	?

How splitting approaches work?

1. Find the best split to perform a splitting approach
2. After, splitting, we obtain the User-item rating matrix
3. Apply any traditional Recommender Algorithm

Context-Aware Splitting Approaches

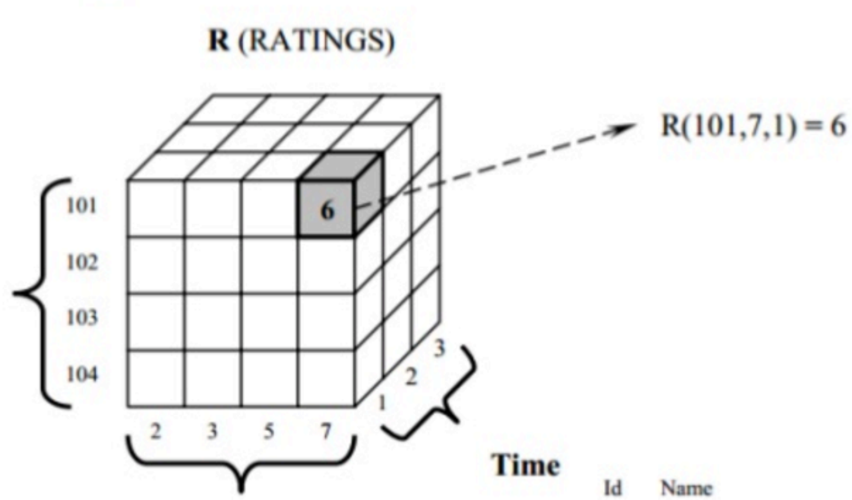
- **Summary:**
  - Pros: Context are fused into user and/or item dimensions
  - Cons: We create new user/items, which increase the sparsity
- **A solution/suggestion:**
  - Create a hybrid recommender to alleviate the data sparsity or cold-start problem introduced by UI Splitting

Contextual Modeling

Independent Contextual Modeling

- Tensor Factorization

  Multi-dimensional space: *User x Items x Contexts -> Ratings*



User

Id	Name	Age
101	John	25
102	Bob	18
103	Alice	27
104	Mary	21

Item

Id	Name	Cost
2	AB17	250.00
3	AB23	299.95
5	XY70	150.00
7	ZZ55	115.50

Time

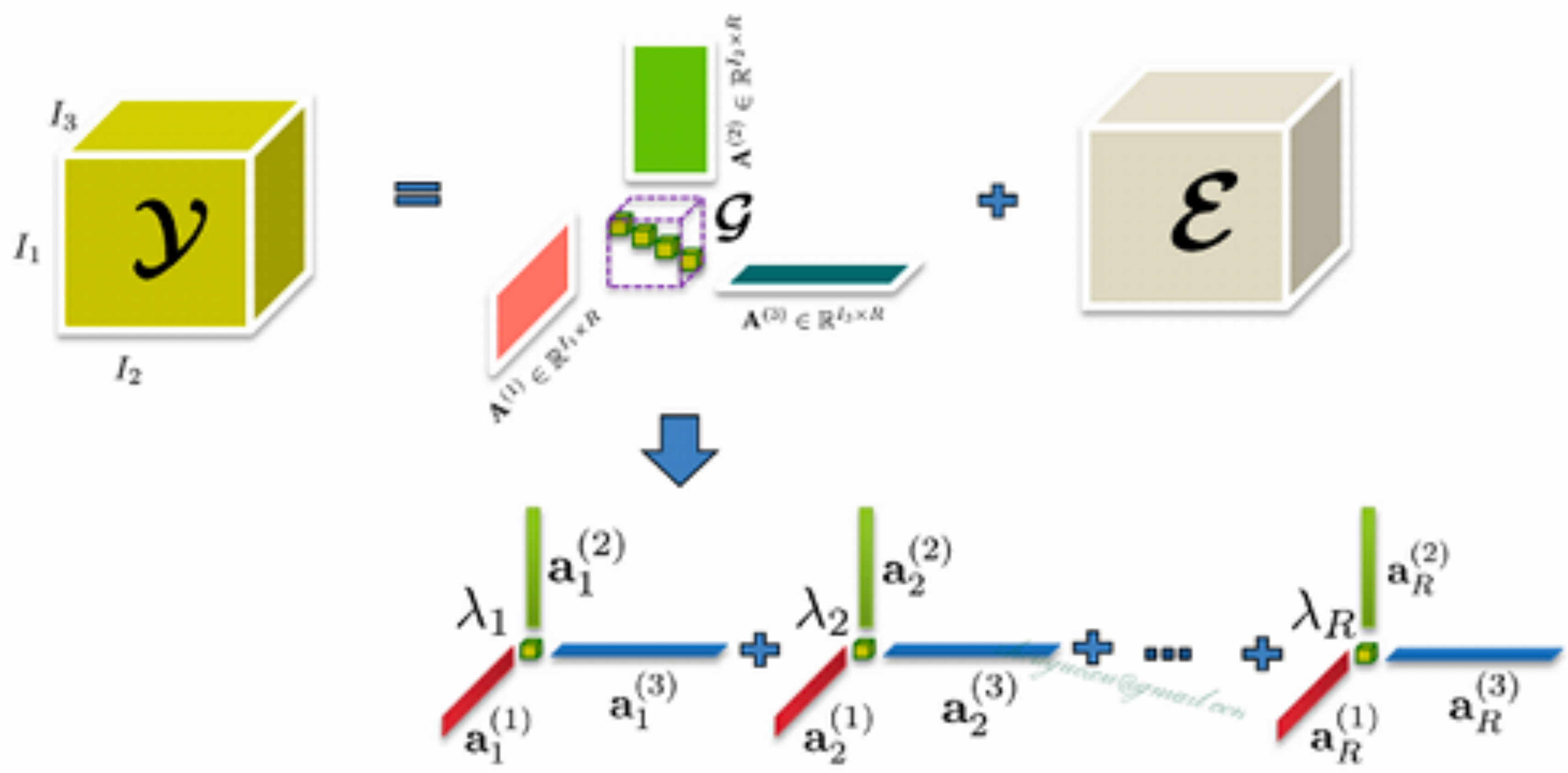
Id	Name
1	Weekday
2	Weekend
3	Holiday

Each context variable is modeled as an individual and independent dimension in addition to user and & item dims

Thus we can create a multidimensional space, where rating is the value in the space

Independent Contextual Modeling

By CANDECOMP/PARAFAC Decomposition



- $R_1 = R_2 = R_3 = R$.
- $\mathcal{G}$ is super diagonal.

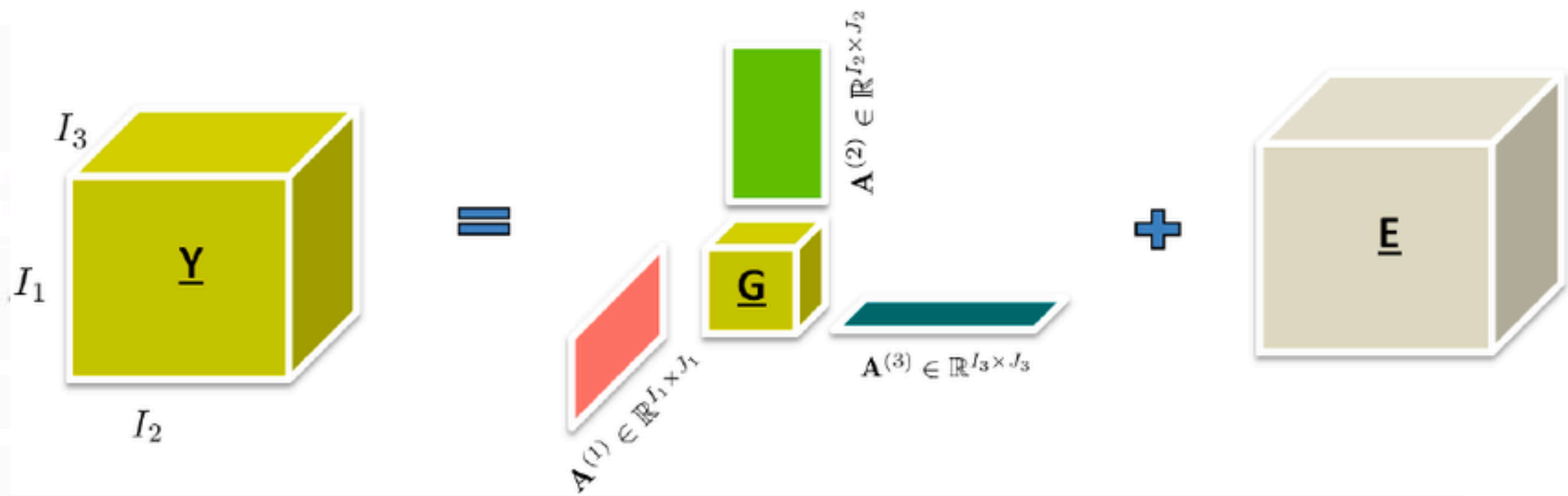
$$\mathcal{Y} = \sum_r \lambda_r \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \mathbf{a}_r^{(3)} + \mathcal{E}$$

UNIVERSITAT DE BARCELONA

DAS

Independent Contextual Modeling

By Tucker Decomposition: Generalization of the SVD. Decompose a tensor into a core tensor, multiplied by a matrix along each mode



$$\boxed{\mathcal{Y} = \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 \mathbf{A}^{(3)} + \mathcal{E}}$$

Independent Contextual Modeling

- Tensor Factorization:
  - **Pros**: Straightforward, easily to incorporate contexts into the model
  - **Cons**:
    - 1) Ignore the dependence between contexts and user/items dims
    - 2) Increased computational cost if more context dimensions

Dependent Contextual Modeling

- **Dependence between Users/Items and Contexts**
  - Users and Context, such as user splitting
  - Item and Context, such as item splitting

  For example if a user can be splitted by time (weekend or not) it tells this user is dependent with this context.

- **Dependence between every two context**
  - Deviation-Based: rating deviation between two contexts
  - Similarity-Based: similarity of rating behaviors in two contexts

Deviation-Based Contextual Modeling

- Contextual Rating Deviation (CRD)
  - How user's ratings is deviated from context c1 to c2?

Context	D1: Time	D2: Location
c1	Weekend	Home
c2	Weekday	Cinema
CRD(Di)	0.5	-0.1

  - CRD(D1) = 0.5 —> Users' rating in Weekday is generally higher than users' rating and Weekend by 0.5
  - CRD(D2) = -0.1 —> Users' rating in Cinema is generally lower than users' rating at home by 0.1

Deviation-Based Contextual Modeling

- Contextual Rating Deviation (CRD)
  - How user's ratings is deviated from context c1 to c2?

Context	D1: Time	D2: Location
c1	Weekend	Home
c2	Weekday	Cinema
CRD(Di)	0.5	-0.1

  - Assume Rating(U, I, c1) = 4
  - Predicted Rating (U, T, c2) = Rating(U, I , c1) + CRDs =

    = 4 + 0.5 - 0.1 = 4.4

Deviation-Based Contextual Modeling

- Build a deviation-based contextual modeling approach
  - Assume ∅ is a special situation: without considering context

Context	D1: Time	D2: Location
Ø	UnKnown	UnKnown
c2	Weekday	Cinema
CRD(Di)	0.5	-0.1

  - Assume Rating(U, I, ∅) = Rating (U,I) = 4
  - Predicted Rating (U, T, c2) = 4 + 0.5 - 0.1 = 4.4

$$F(U, I, C) = P(U, I) + \sum_{i=0}^{N} CRD(i)$$

Deviation-Based Contextual Modeling

- Build a deviation-based contextual modeling approaches:
  - Simplest method: $$F(U,I,C) = P(U,I) + \sum_{i=0}^{N} CRD(i)$$
  - User-personalized model: $$F(U,I,C) = P(U,I) + \sum_{i=0}^{N} CRD(i,U)$$
  - Item-personalized model: $$F(U,I,C) = P(U,I) + \sum_{i=0}^{N} CRD(i,I)$$