Content-Based Recommenders

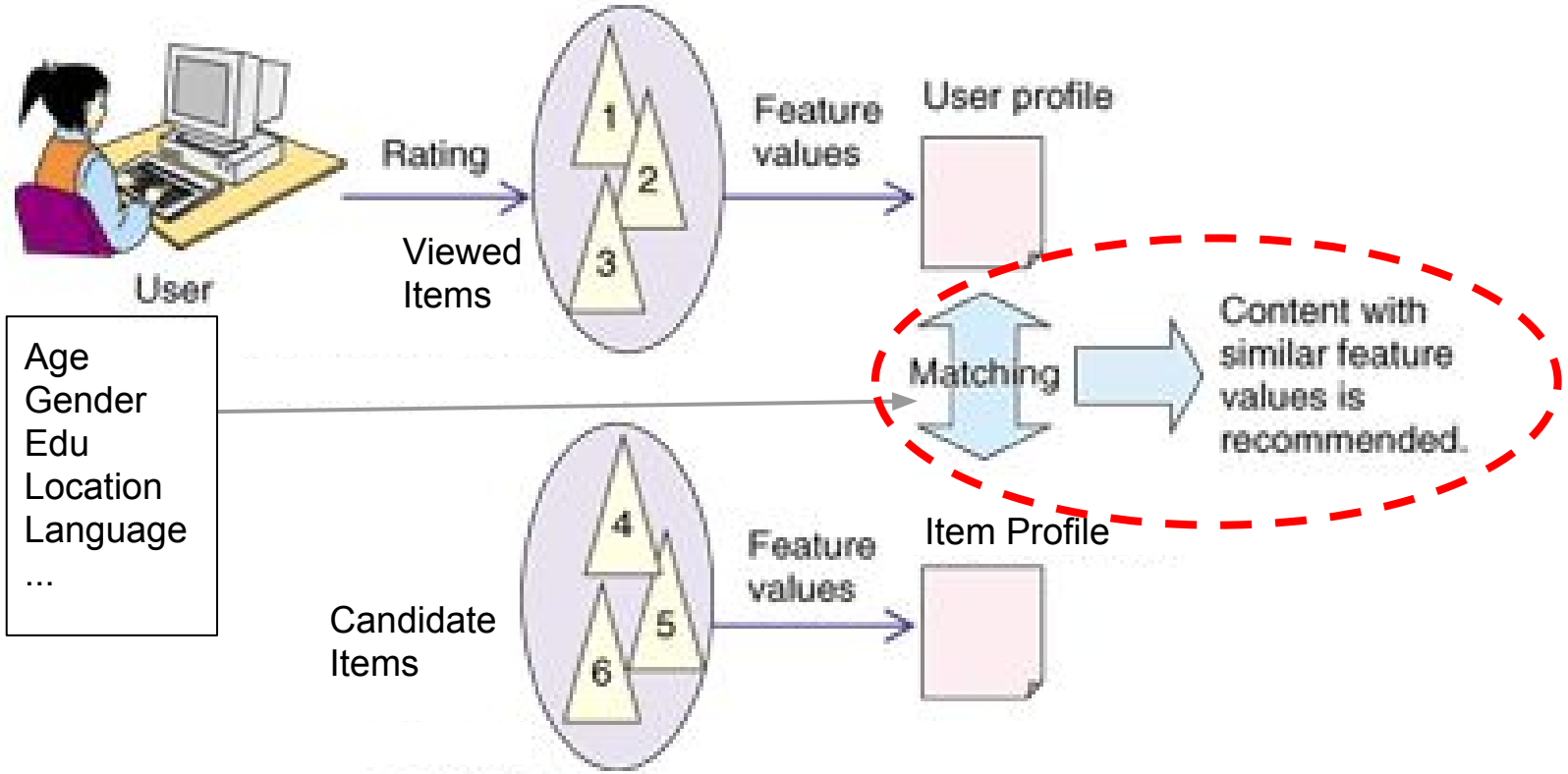
David Yang
xdyang70@gmail.com

Key Idea

Recommend products to a customer similar to previous items rated highly by him/her.

Examples:

- Song: Genre, Style, Artist, Mood...
- Clothing: Style, Brand, Color ...
- Article: Topic, content


Rating
Viewed
Items
1
2
3
Feature
values
User profile
User
Age
Gender
Edu
Location
Language
...
Matching
Content with
similar feature
values is
recommended.
Candidate
Items
4
5
6
Feature
values
Item Profile

Item Profile (Item Feature Vector)

Profile - A Set of Features Describing the Items

- Text (Blog & Article): bag of words, lda, word2vec
- Audio (Music & Song): time/frequency domains
- Image (Photo and Video): wide & deep network

A Vector: Sparse (symbolic) v.s. Dense (semantic)

User Profile (User Feature Vector)

Endogenous (Customer Segmentation):

- Taste or interest self-reported by the user
- Age, Gender, Education, Location, Language, Device….

Exogenous (Aggregate Item Features):

- Simple: Normalized, Weighted Average
- Sophisticated: PCA, latent factors

A Simple Illustration

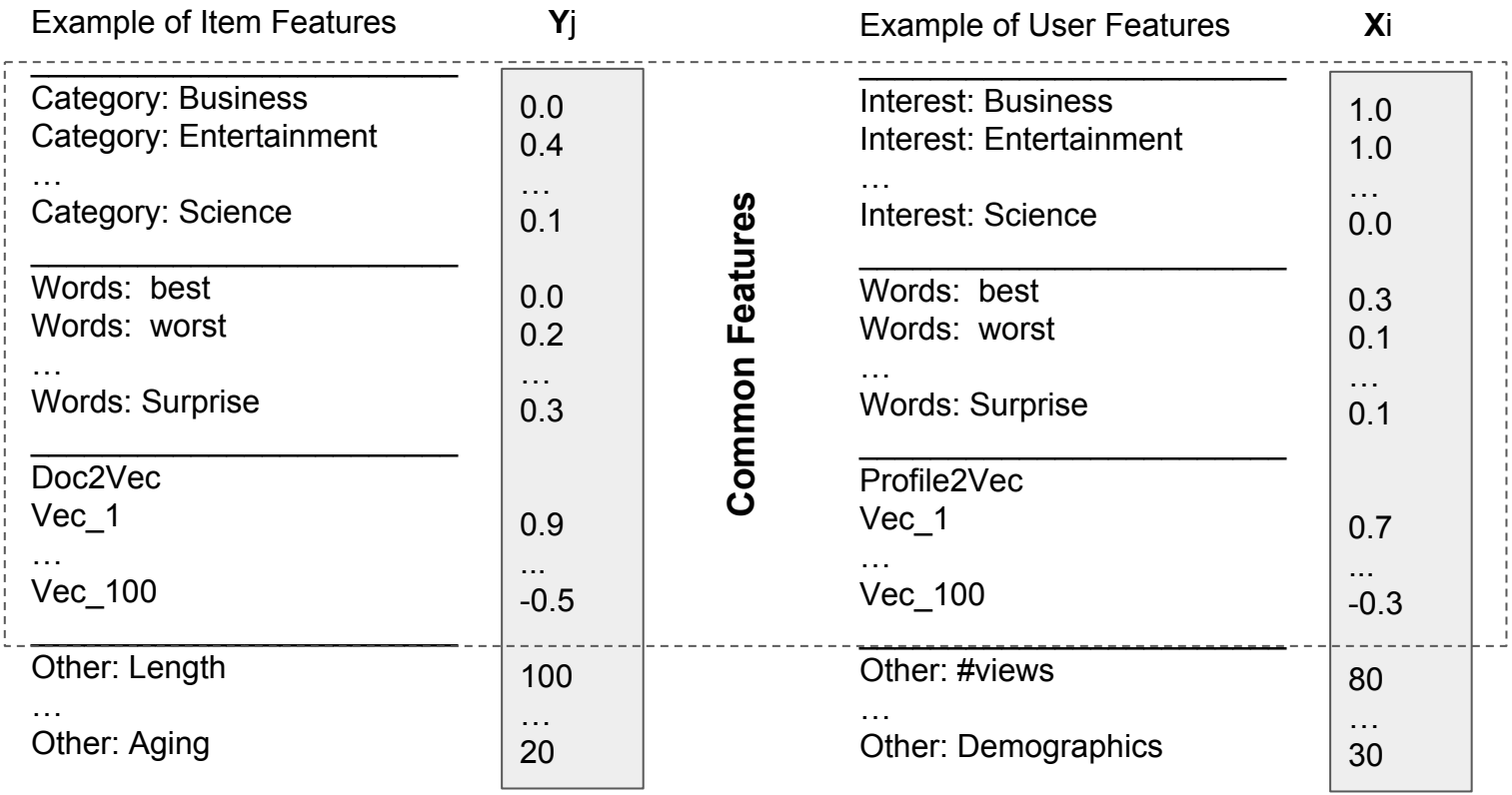
A user rated 5 movies directed by Bob and Mary

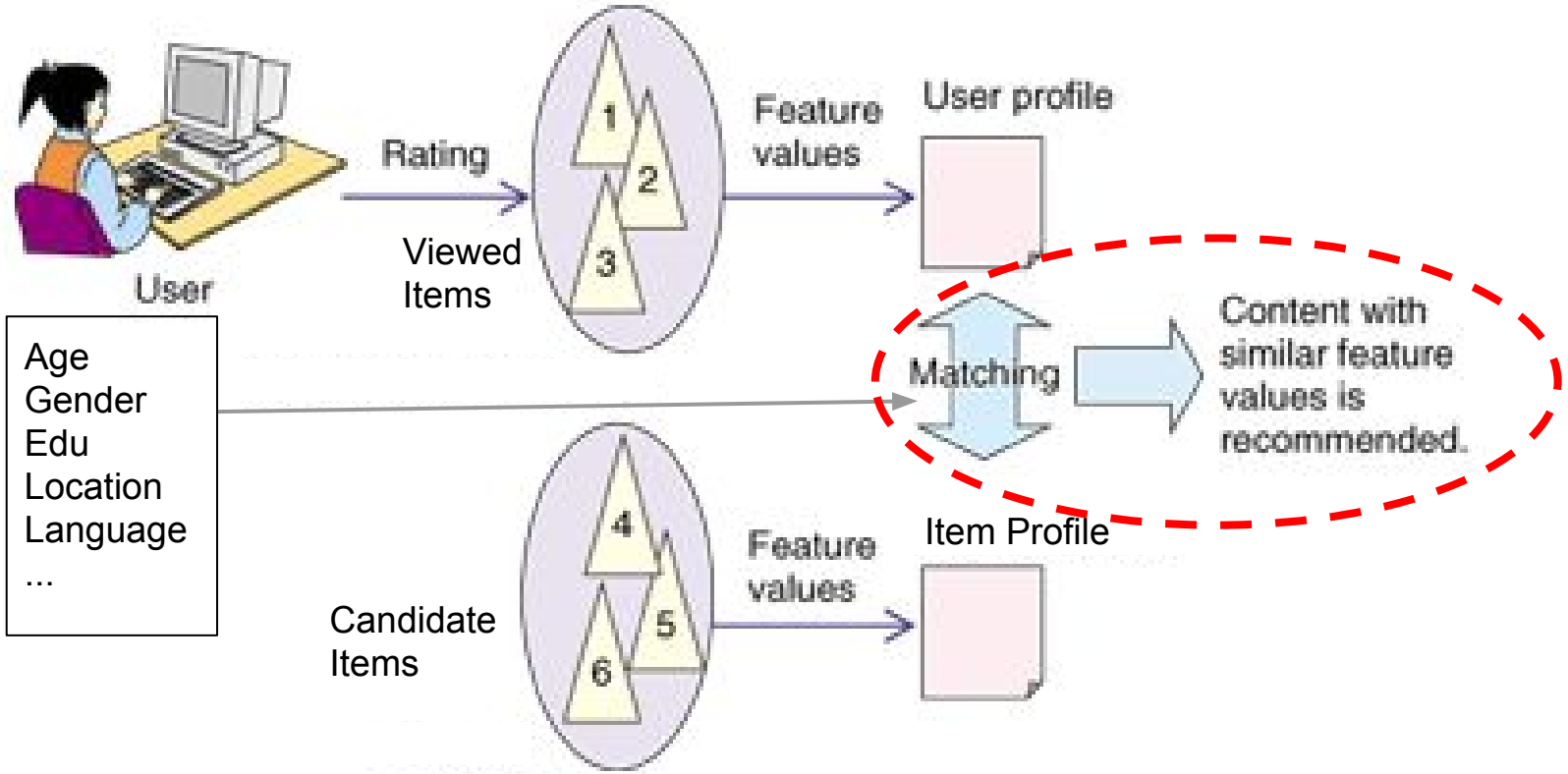
	Movie 1 (Bob)	Movie 2 (Bob)	Movie 3 (Mary)	Movie 4 (Mary)	Movie 5 (Mary)	Mean Rating
Rating (1-5 star)	**3**	**5**	**1**	**2**	**4**	**3**

Then, his/her profile (reflecting director preference)

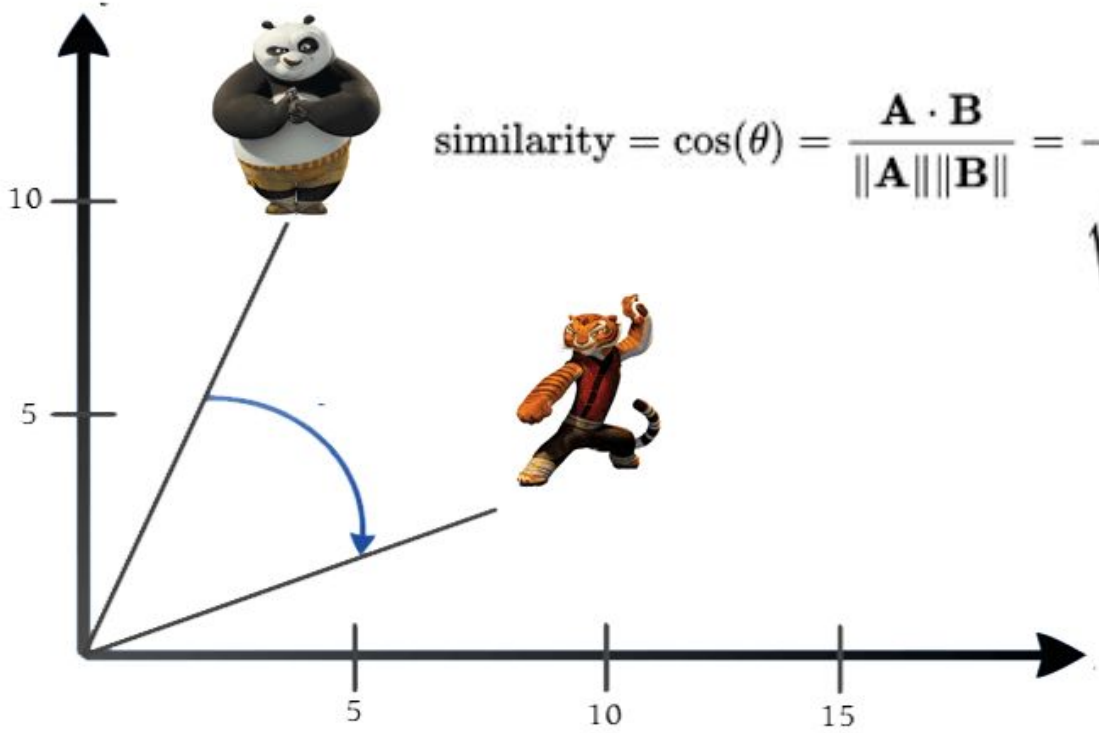
(Bob, Mary) = (4, 2.3) OR (1, -0.66) [with normalization]

A More Realistic Illustration



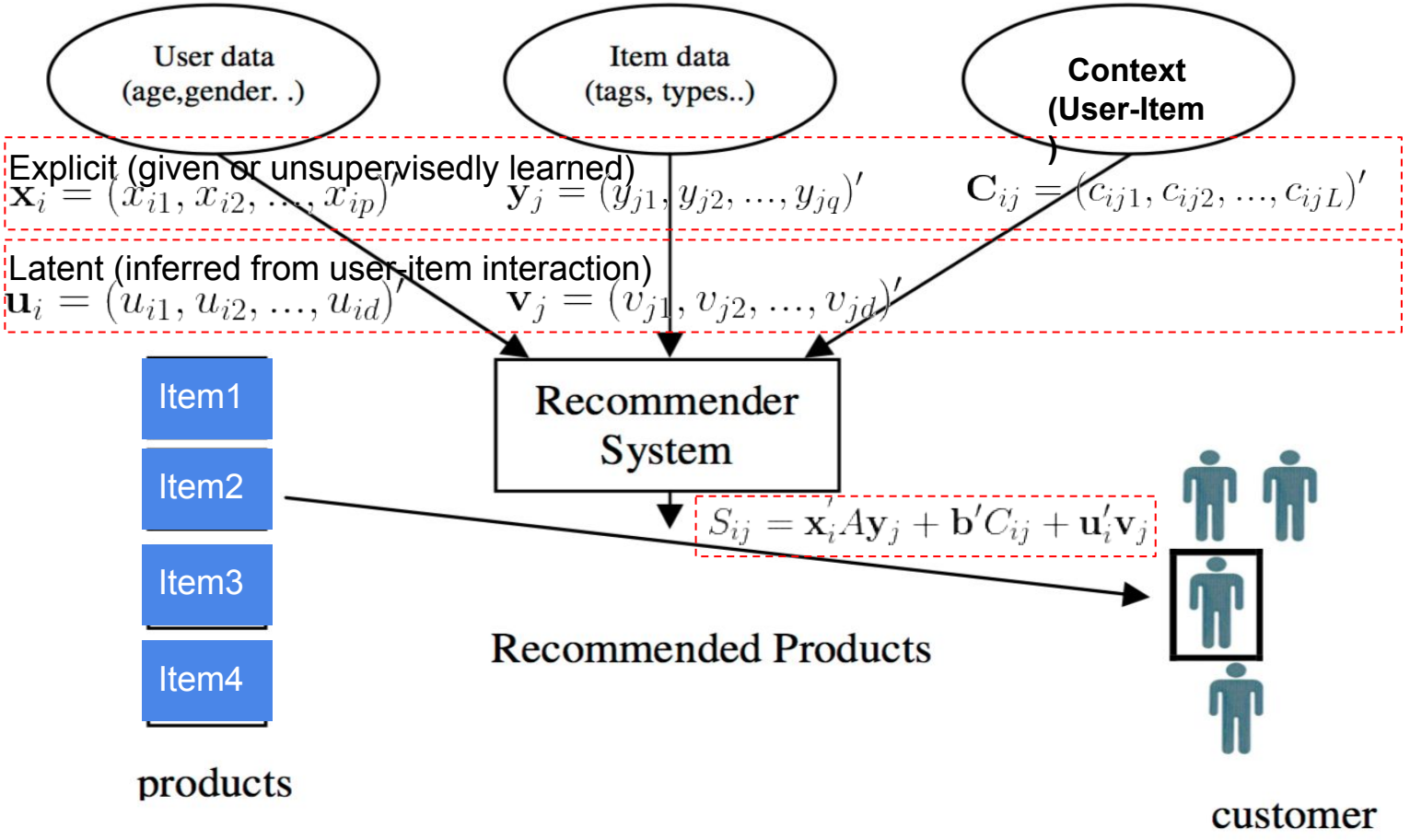

Rating
Viewed
Items
1
2
3
Feature
values
User profile
User
Age
Gender
Edu
Location
Language
...
Matching
Content with
similar feature
values is
recommended.
Candidate
Items
4
5
6
Feature
values
Item Profile

Matching User and Item in Feature Vector Space



$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

Matching User and Item Using Predictive Model



Summary of Content-Based Recommenders

Pros

- Simple computing with Intuitive Explanation: (1) do not need data on other users; (2) tailor to unique taste of a user
- Handle “Cold Start”: (1) able to recommend new or unpopular items; (2) able to recommend items to new customers (if endogenous features are give)

Cons

- Sometimes, hard to find or extract proper features.
- Overspecification (items outside user’s content profile?) & unable to use other user’s data (e.g., judgement)
- “Cold Start’: new user has no historical items to derive endogenous features.

Example 1

In this example, we will learn how to use Python to extract features from text in describing scientific papers, and build a simple content based recommender system to recommend papers to read.

(see Jupyter Notebook: “Example 1 Content-based Recommender with TF-IDF.ipynb”)