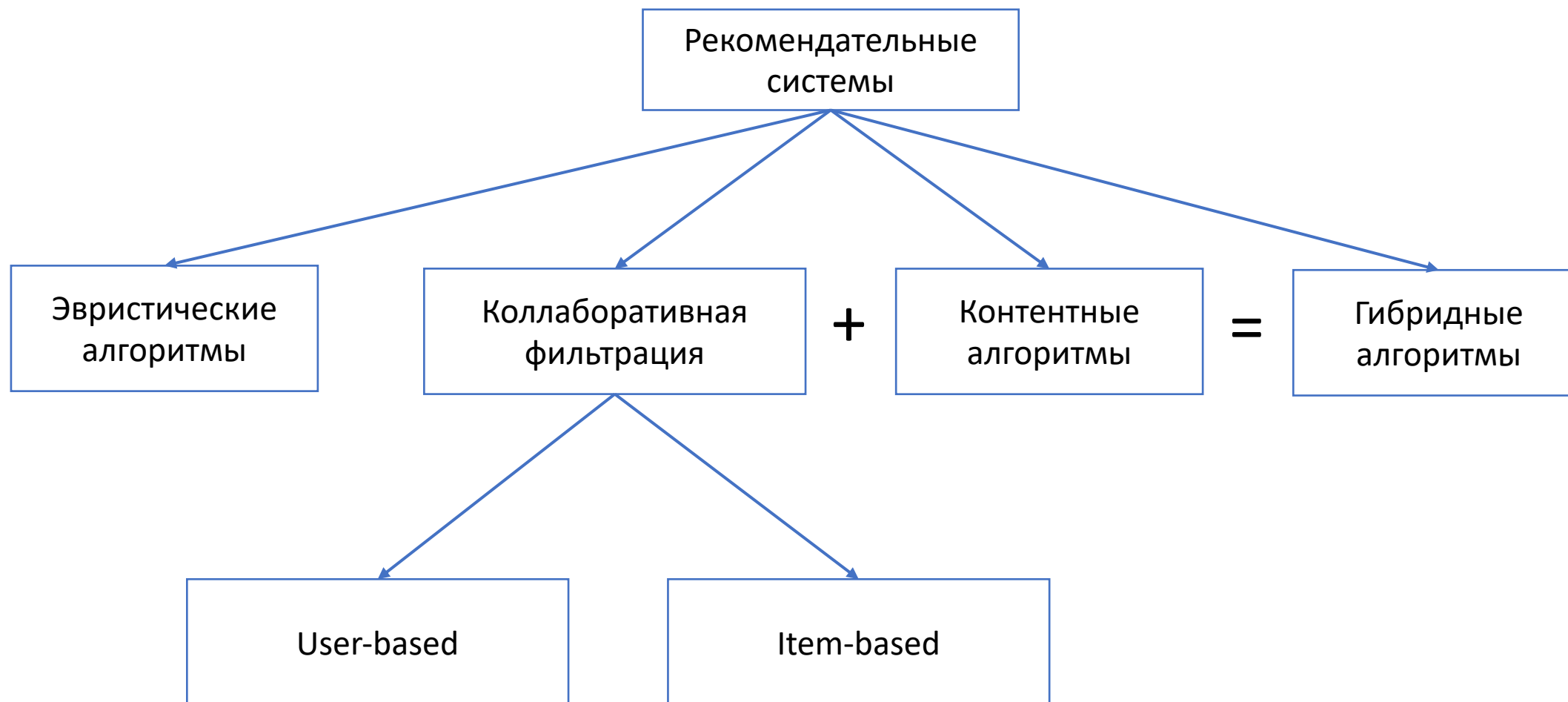


# Классификация алгоритмов RecSys



# Эвристические алгоритмы



Popular recommender

Рекомендует всем пользователям самые популярные товары

# Эвристические алгоритмы



## Popular recommender

Рекомендует всем пользователям самые популярные товары

### Плюсы:

- ✓ Простой
- ✓ Может давать неплохое качество

### Минусы:

- ✓ Не персонализированный
- ✓ Не рекомендует большую часть товаров

# Эвристические алгоритмы



Random recommender

Рекомендует всем пользователям случайные товары

# Эвристические алгоритмы



## Random recommender

Рекомендует всем пользователям случайные товары

### Плюсы:

- ✓ Простой
- ✓ Рекомендации покрывают почти все объекты

### Минусы:

- ✓ Не учитывает потребности пользователя
- ✓ Качество, как правило, довольно низкое

# Эвристические алгоритмы



Random Popular weighted recommender

Рекомендует всем пользователям случайные товары с вероятностью, пропорциональной популярности

# Эвристические алгоритмы



## Random Popular weighted recommender

Рекомендует всем пользователям случайные товары с вероятностью, пропорциональной популярности

### Плюсы:

- ✓ Простой
- ✓ Рекомендации покрывают почти все объекты
- ✓ Лучше учитывает популярность объектов

### Минусы:

- ✓ Не учитывает потребности пользователя
- ✓ Качество, как правило, довольно низкое, но лучше чем у Random

# Эвристические алгоритмы



User Popular recommender

Рекомендует товары, которые каждый пользователь чаще всего покупал



# Эвристические алгоритмы



## User Popular recommender

Рекомендует товары, которые каждый пользователь чаще всего покупал

### Плюсы:









- ✓ Простой
- ✓ Показывает хорошее качество для регулярных покупок

### Минусы:

- ✓ Не рекомендует пользователю новые товары

# Взаимодействия User/Item



						
	4				5	
	4	5		5		
		5		4		3
-----						
 Lizzy	5	4	?	?	?	?

Historical data

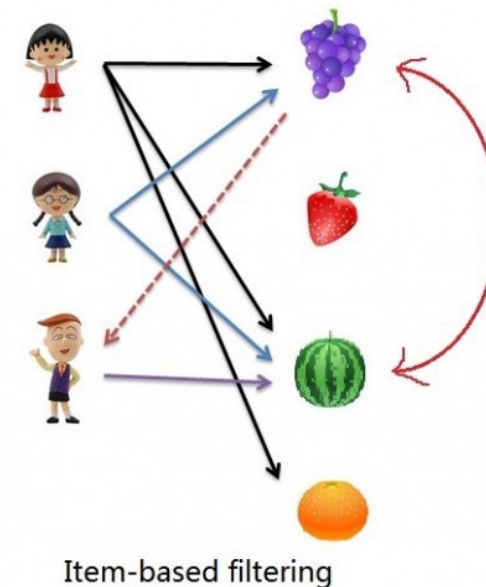
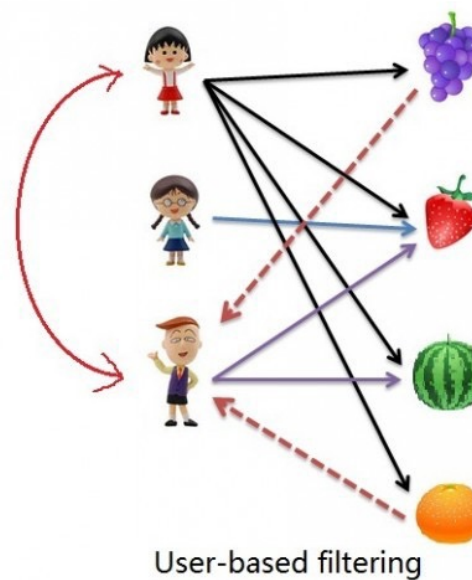
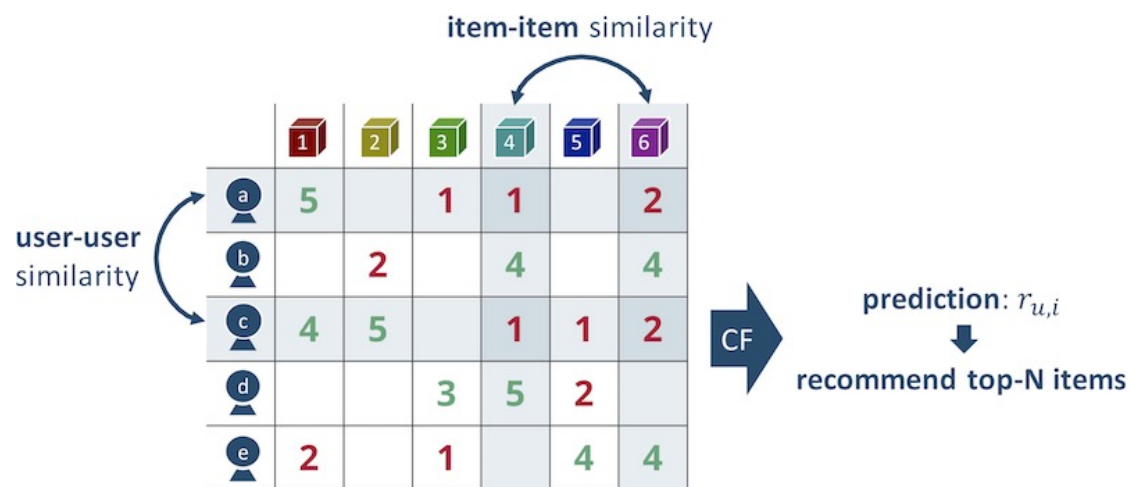
New user

В ячейках матрицы может быть

- Рейтинг
- Факт покупки (0/1)
- Количество покупок
- Взвешенная метрика

+ 0.1 \* кол-во просмотров  
+ 0.3 \* кол-во лайков  
+ 0.5 \* кол-во комментариев  
+ 0.9 \* кол-во покупок

# User/Item based алгоритмы



Часто могут показывать достаточно высокое качество

# Item based алгоритм



Необходимо построить матрицу похожести объектов

Есть несколько способов ее формировать

1. Скалярное произведение

$$\text{simil}(x, y) = \cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\| \times \|\vec{y}\|} = \frac{\sum_{i \in I_{xy}} r_{x,i} r_{y,i}}{\sqrt{\sum_{i \in I_x} r_{x,i}^2} \sqrt{\sum_{i \in I_y} r_{y,i}^2}}$$

2. Косинусная близость

$$TF(t, d) = \frac{\text{number of times } t \text{ appears in } d}{\text{total number of terms in } d}$$

$$IDF(t) = \log \frac{N}{1 + df}$$

3. Взвешивание с весом tf-idf (user=term, item=doc )

$$TF - IDF(t, d) = TF(t, d) * IDF(t)$$

4. Взвешивание с весов bm25

$$\text{score}(D, Q) = \sum_{i=1}^n \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{\text{avgdl}})},$$