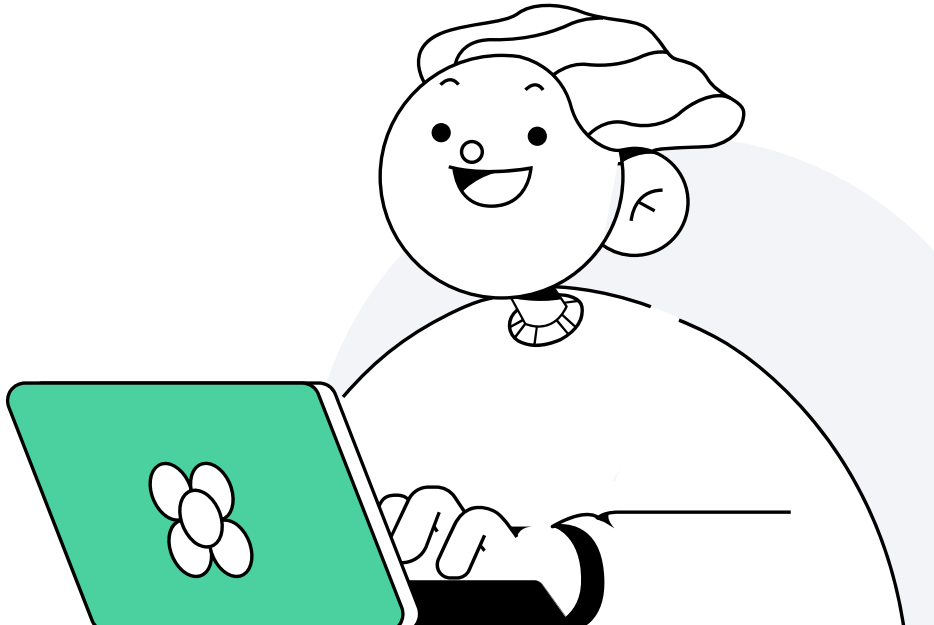
Коллаборативная фильтрация

План занятия

1. Что такое коллаборативная фильтрация
2. Item-based коллаборативная фильтрация
3. User-based коллаборативная фильтрация
4. Пакет Surprise

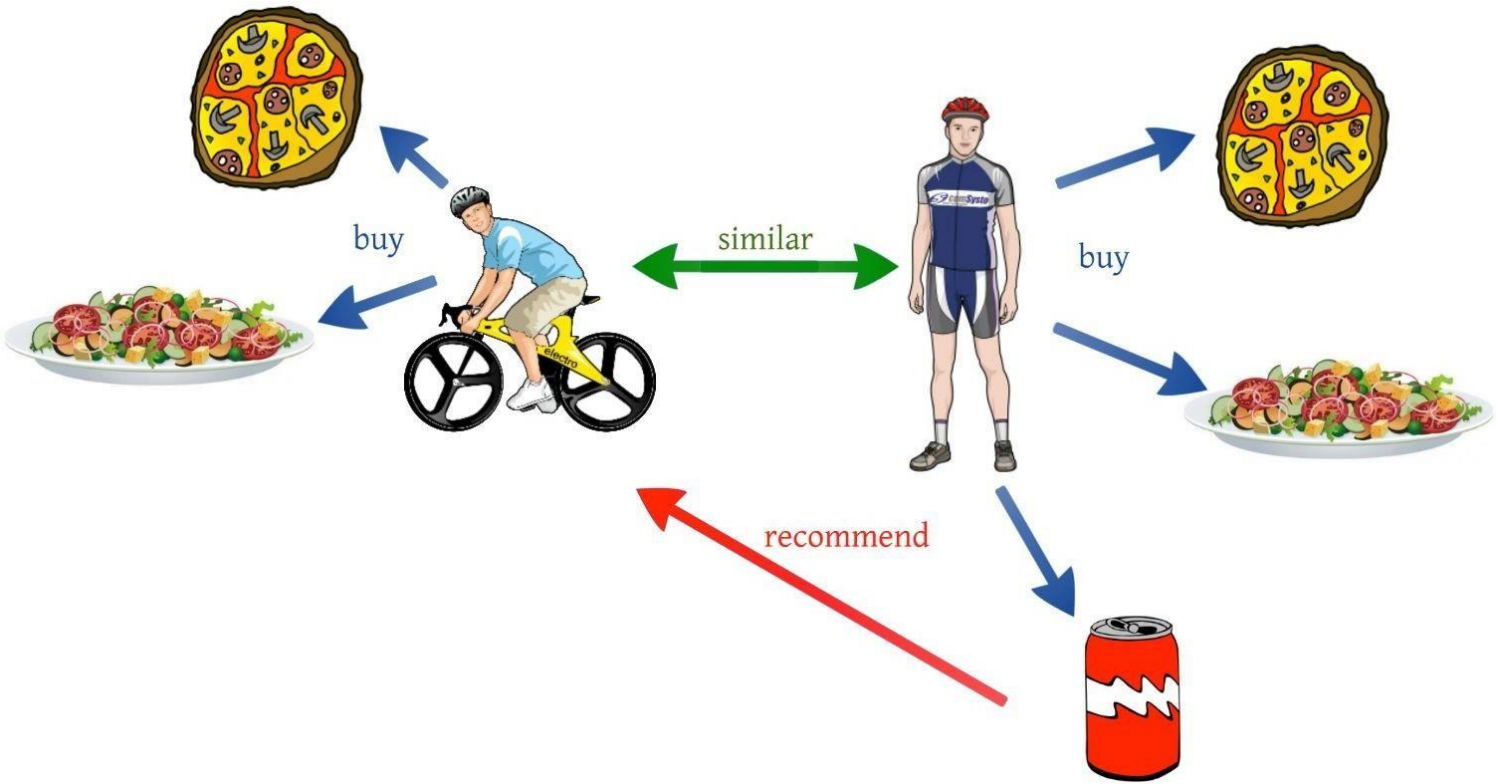


Что такое коллаборативная фильтрация

1

Фильтрация по содержанию

Коллаборативная фильтрация



Коллаборативная фильтрация

Используются только результаты взаимодействий (рейтинг, покупка и др.)

	item_1	item_2	...	item_m
user_1	2	0	...	1
	...	...	...	...
user_n	5	2	...	0

Плюсы

- Не нужно налаживать процесс извлечения фичей
- Мало зависит от предметной области
- Предсказываем поведение и обучаемся на нём же

Минусы

- Проблема холодного старта (новый юзер/айтем - не ясно, что рекомендовать)
- Много данных нужно

Предсказания

Предсказать значения потенциального взаимодействия

	item_1	item_2	...	item_m
user_1	2	?	...	1
...	...	...	...	...
user_n	5	2	...	?

Item-to-item collaborative filtering

2

Как кодируются объекты

Айтемы описываются результатами взаимодействия со всеми пользователями

$$i_1 = \left(r_{u_1}\left(i_1\right), r_{u_2}\left(i_1\right), \ldots, r_{u_N}\left(i_1\right)\right)$$

$$i_2 = \left(r_{u_1}\left(i_2\right), r_{u_2}\left(i_2\right), \ldots, r_{u_N}\left(i_2\right)\right)$$

Расстояние между объектами

Ищем похожести айтемов по метрике расстояния

$$d^2\left(i_1, i_2\right) = \sum_u \left(r_u\left(i_1\right) - r_u\left(i_2\right)\right)^2$$

Какие бывают расстояния

- Евклидово расстояние
- Косинусное расстояние
- Манхэттенское расстояние

Какие бывают расстояния

Могут использоваться и семантические метрики (взаимодействие рассматривается как последовательность)

- Расстояние Хэмминга
- Расстояние Левенштейна
- Коэффициент Жаккара

Практика

Item-to-item

Гипотеза: показывать под фильмами похожие на них другие фильмы.

Что делать

1. Найдем векторы фильмов
2. Найдем 10 похожих на него

User-based collaborative filtering

Алгоритм

Есть матрица оценок, выставленных пользователями продуктам

	1	2	3	4	5	6	7	8	9
alex	5.0000	3.0000			4.0000				
ivan	4.0000					1.0000		2.0000	3.0000
bob		5.0000	5.0000						
david			4.0000	3.0000		2.0000	1.0000		

Алгоритм

- Выбрать K пользователей, предпочтения которых больше всего похожи на вкусы рассматриваемого юзера
- Похожесть измеряем стандартными метриками
- Для каждого юзера умножаем его оценки на вычисленную величину меры
- Получаем взвешенные оценки по айтемам для рассматриваемого юзера

	alex	bob	david	**sum**
ivan	0.5164	0.0000	0.0667	**0.5831**

Алгоритм

- Для каждого айтема считаем сумму калиброванных оценок
- Полученное значение делим на сумму мер близких пользователей
- К примеру, для alex 5*0.5164=2.582, 3*0.5164=1.5492 и т.д.
- result_2 = 1.5492/0.5831

	1	2	3	4	5	6	7	8	9
alex	2.5820	1.5492	0.0000	0.0000	2.0656	0.0000	0.0000	0.0000	0.0000
bob	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
david	0.0000	0.0000	0.2668	0.2001	0.0000	0.1334	0.0667	0.0000	0.0000
sum	2.5820	1.5492	0.2668	0.2001	2.0656	0.1334	0.0667	0.0000	0.0000
result	4.4281	**2.6568**	**0.4576**	**0.3432**	**3.5424**	0.2288	**0.1144**	0.0000	0.0000

Особенности

- Лучше работает, когда объектов больше, чем пользователей

Item-based collaborative filtering

4

Алгоритм

Есть матрица оценок, выставленных юзерами фильмам

	Вася	Петр	Валера	Жанна	Петрович
Трактористы	?	3	5	5	2
Свинарка и пастух	3	5	3	5	3
Once upon a Tractor	4	2		5	
Tractor, Love, pigeon	5	2	4		2
Babe	2	5	3	4	2

Алгоритм

- Люди ведут себя по-разному
- В этом случае вычтем из каждого вектора оценок среднюю оценку каждого пользователя

К примеру, у Васи средняя оценка (3+4+5+2)/4=3.5

	Вася	Петр	Валера	Жанна	Петрович
Трактористы	?	3	5	5	2
Свинарка и пастух	3	5	3	5	3
Once upon a Tractor	4	2		5	
Tractor, Love, pigeon	5	2	4		2
Babe	2	5	3	4	2
Среднее	3,5	3,4	3,75	4,75	2,25

Алгоритм

- Люди ведут себя по-разному
- В этом случае вычтем из каждого вектора оценок среднюю оценку каждого пользователя

К примеру, у Васи средняя оценка (3+4+5+2)/4=3.5

	Вася	Петр	Валера	Жанна	Петрович
Трактористы	?	-0,4	1,25	0,25	-0,25
Свинарка и пастух	-0,5	1,6	-0,75	0,25	0,75
Once upon a Tractor	0,5	-1,4		0,25	
Tractor, Love, pigeon	1,5	-1,4	0,25		-0,25
Babe	-1,5	1,6	-0,75	-0,75	-0,25

*Этот подход могли применить и к User-based

Алгоритм

Для фильма *«Трактористы»* считаем любую метрику похожести к выбранному фильму

	Item-Based корреляция
Свинарка и пастух	-0,9545
Once upon a Tractor	1
Tractor, Love, pigeon	0,787
Babe	-0,6689
Сумма по модулю	3,4104

Алгоритм

- Так же как и в случае с user-based, считаем взвешенное среднее, но для уже оценённых юзерами фильмов
- И делим на сумму похожестей (3.1611/3.4101)
- На нашем примере подход item-based предполагает, что Вася поставит *«Трактористам»* оценку 4.4 (3.5 + 0.92)

	Вася
Трактористы	?
Свинарка и пастух	-0,5
Once upon a Tractor	0,5
Tractor, Love, pigeon	1,5
Babe	-1,5

	Item-Based корреляция
Свинарка и пастух	-0,9545
Once upon a Tractor	1
Tractor, Love, pigeon	0,787
Babe	-0,6689
Сумма по модулю	3,4104

	Вася
Трактористы	?
Свинарка и пастух	0,47725
Once upon a Tractor	0,5
Tractor, Love, pigeon	1,1805
Babe	1,00335
Сумма	3,1611
Результат	0,9269000704
Оценка	4,42690007

Особенности

- Лучше работает, когда пользователей больше, чем объектов

Практика

User-based и Item-based

Задача: рекомендации на главной странице сервиса в разделе «Персональная подборка».

Что делать

1. Использовать алгоритмы из Surprise, основанные на kNN
2. Взять любого пользователя и посмотреть на результаты предсказаний

Коллаборативная фильтрация