НЕТОЛОГИЯ
групп

КОЛЛАБОРАТИВНАЯ ФИЛЬТРАЦИЯ

Артур Сапрыкин

ПЛАН ЗАНЯТИЯ

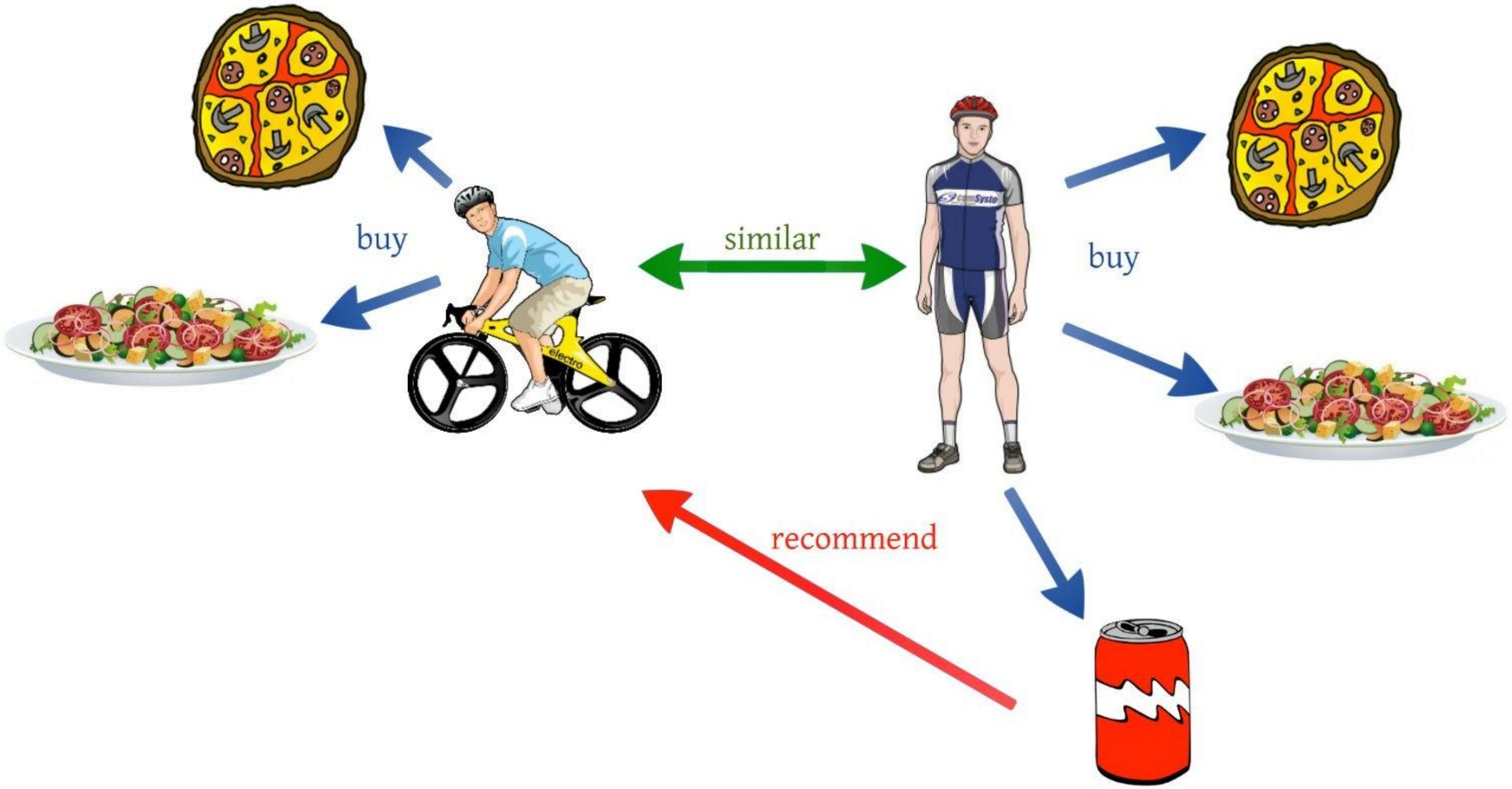
ПЛАН ЗАНЯТИЯ

1. расстояния в рекомендациях
2. item-based коллаборативная фильтрация
3. user-based коллаборативная фильтрация
4. пакет surprise

ЧТО ТАКОЕ КОЛЛАБОРАТИВНАЯ ФИЛЬТРАЦИЯ

ФИЛЬТРАЦИЯ ПО СОДЕРЖАНИЮ

КОЛЛАБОРАТИВНАЯ ФИЛЬТРАЦИЯ



ПЛЮСЫ

- не нужно налаживать процесс извлечения фичей
- особо не зависит от предметной области
- предсказываем поведение и обучаемся на нём же

МИНУСЫ

- сложно объяснить рекомендации
- проблема холодного старта
- много данных

ITEM-TO-ITEM COLLABORATIVE FILTERING

КАК КОДИРУЮТСЯ ОБЪЕКТЫ

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

КАК КОДИРУЮТСЯ ОБЪЕКТЫ

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

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

КАК ВЫЧИСЛИТЬ РАССТОЯНИЕ

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

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

РАССТОЯНИЕ МЕЖДУ ОБЪЕКТАМИ

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

КАКИЕ БЫВАЮТ РАССТОЯНИЯ

- евклидово расстояние
- косинусное расстояние
- манхэттенское расстояние

КАКИЕ БЫВАЮТ РАССТОЯНИЯ

- расстояние Хэмминга
- коэффициент Жаккара
- коэффициент Танимото

ПРАКТИКА

ПРАКТИКА

ITEM-TO-ITEM

Гипотеза - показывать под фильмами “похожие” на него другие фильмы. Метрику похожести хотим протестировать на основе поведения пользователей

Что делать?

1. Найдите свой любимый фильм
2. Найдите 10 похожих на него
3. Попробуйте разные метрики схожести

Сколько есть времени?

15 минут

USER-BASED COLLABORATIVE FILTERING

АЛГОРИТМ

- имеется 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
- “похожесть” измеряем стандартными метриками (например, косинусное расстояние)
- для каждого пользователя умножаем его оценки на вычисленную величину меры

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

АЛГОРИТМ

- для каждого из продуктов считаем сумму калиброванных оценок, полученное значение делим на сумму мер близких пользователей

	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
- может использоваться для объяснения рекомендаций (социальная составляющая)

ПРАКТИКА

ПРАКТИКА

USER-BASED

Задача - рекомендации на главной странице сервиса в разделе “Персональная подборка”

Что делать?

1. Датасет тот же ml-latest
2. Использовать алгоритмы из surprise, основанные на KNN, флаг user_based ставим True
3. Взять любого пользователя и посмотреть на результаты предсказаний

Сколько есть времени?

15 минут

ITEM-BASED COLLABORATIVE FILTERING

АЛГОРИТМ

- имеется матрица оценок, выставленных пользователями фильмам

	Трактористы	Свинарка и пастух	Once Upon a Tractor	Tractor, Love & Rock'n Roll	Babe
Вася	?	3	4	5	2
Пётр	3	5	2	2	5
Валерик	5	3		4	3
Жанночка	5	5	5		4
Петрович	2	3		2	2

АЛГОРИТМ

- люди ведут себя по-разному, поэтому в этом случае вычтем из каждого вектора оценок среднюю оценку каждого пользователя

	Трактористы	Свинарка и пастух	Once Upon a Tractor	Tractor, Love & Rock'n Roll	Babe
Вася	?	−0.5	0.5	1.5	−1.5
Пётр	−0.4	1.6	−1.4	−1.4	1.6
Валерик	1.25	−0.75		0.25	−0.75
Жанночка	0.25	0.25	0.25		−0.75
Петрович	−0.25	0.75		−0.25	−0.25

АЛГОРИТМ

- для каждого фильма считаем коэффициент корреляции Пирсона к выбранному фильму (или любую другую метрику расстояния)

	Item-based корреляция
Свинарка и пастух	−0.9545
Once Upon a Tractor	1
Tractor, Love & Rock'n Roll	0.7870
Babe	−0.6689

АЛГОРИТМ

- так же, как и в случае с user-based считаем взвешенное среднее, но для уже оцененных пользователем фильмов
- на нашем примере item-based подход предполагает, что Вася поставит “Трактористам” оценку 4.4

ОСОБЕННОСТИ

- лучше работает, когда пользователей больше, чем объектов
- эффективно работает в реальном времени
- хорошо работает даже на больших данных

ПРАКТИКА

ПРАКТИКА

ITEM-BASED

Задача - рекомендации на главной странице сервиса в разделе “Персональная подборка”

Что делать?

1. Датасет тот же ml-latest
2. Использовать алгоритмы из surprise, основанные на KNN, флаг user_based ставим True
3. Поэкспериментировать с метриками, добавить кросс-валидацию

Сколько есть времени?

15 минут

ДОМАШНЯЯ РАБОТА

ПАКЕТ SURPRISE

- используйте данные MovieLens 1M
- можно использовать любые модели из пакета
- получите RMSE на тестовом сете 0.87 и ниже

ВОПРОСЫ