

Заключение про машинное обучение

Виды обучения

1. Обучение с учителем

- a. Регрессия
- b. Классификация
- c. Ранжирование

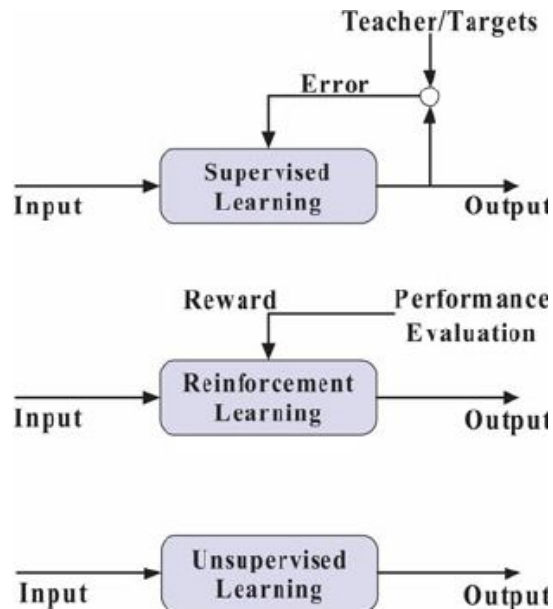
2. Обучение без учителя

- a. Кластеризация
- b. Снижение размерности

3. Обучение с подкреплением

4. Активное обучение, обучение с частичным привлечением учителем

Виды обучения



Спасибо!

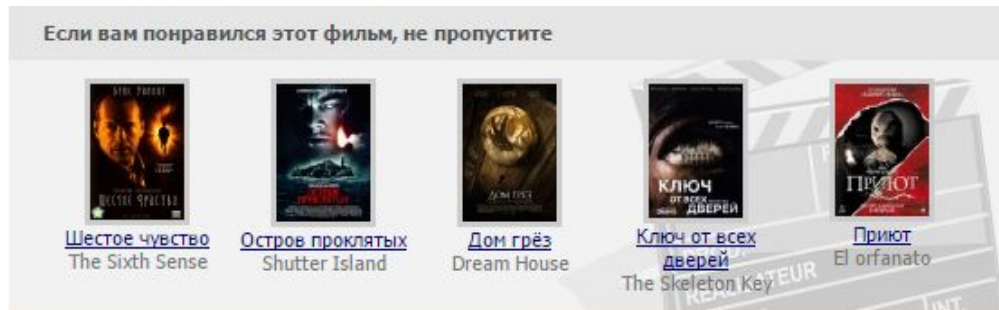
Вопросы?

Введение в рекомендательные системы

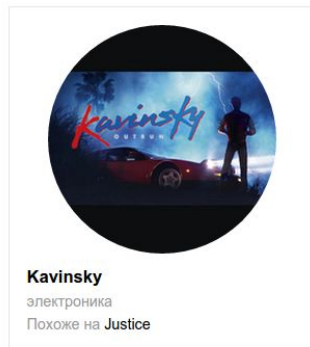
План

1. Что такое рекомендательные системы
2. Классические методы
 - 2.1. Контентные рекомендации
 - 2.2. Коллаборативная фильтрация
 - 2.2.1. User-based, item-based
 - 2.2.2. SVD, matrix factorization
 - 2.3. Другие методы: кластеризация, ассоциативные правила
3. Современные методы
 - 3.1. Классификация/регрессия
 - 3.2. Контекстные рекомендации
 - 3.3. Тензорная факторизация
 - 3.4. Ранжирование
4. Заключение

Примеры рекомендаций




Ищут вместе с исполнителями, которых вы слушаете



Примеры рекомендаций

[Home](#) / [Electronics](#) / [MP3 Players](#) / [MP3 Players](#) / [16GB A Series Walkman Video MP3](#)



16GB A Series Walkman Video MP3

Sony

\$219.00

CODE: H0148VPS1T

Availability: **In stock**

Quantity:


ADD TO CART

FREE US shipping over \$100!
Orders within next 2 days will be shipped on Monday

Description


With an A Series by your side, there's never a dull moment. The dazzling 2.8" (400x240) touch screen delivers incredible color and quickly serves up photos, videos, album artwork and more. Easily navigate your media library with the tap of a finger or via traditional button controls if you prefer. Experience your music wirelessly and stream audio to other compatible devices with integrated Bluetooth®. Sony Clear Audio technologies work as a team to make your music sound the best it can - Digital Sound Enhancement Engine, Clear Stereo and Clear Bass Audio Technologies offer deeper, richer sound plus S-Master™ MX amplification makes for higher signal to noise ratios and less distortion. Upload the matching lyric file and learn the words to your favorite song as your A Series scrolls them along the screen in sync with the music. Premium EX earbuds and USB cable included.

Customers Who Bought This Item Also Bought



adizero Rush Shoes

\$100.00



Extra Bass Headphones - 70mm

\$299.00

Примеры рекомендаций

The screenshot displays the YouTube homepage interface. At the top, the browser address bar shows 'https://www.youtube.com'. The YouTube logo and navigation icons are on the left. A sidebar on the left lists categories: Popular on YouTube, Music, Sports, Gaming, Movies, TV Shows, News, Live, Spotlight, and 360° Video. The main content area features a 'What to Watch' tab and a 'Music' tab. Below these, a large video player shows 'FAN' by Shah Rukh Khan. To the right of the player, there are three smaller video thumbnails: 'FAN - Teaser 1 | Shah Rukh Khan', 'Brock Lesnar destroys J&J Security's prized Cadillac: Raw, July 6, 2015', and 'Salman Khan's SHOCKING Comment on Shahrukh Khan At Bajrangi Bhaijaan Trailer...'. Below the main video player, a 'Recommended' section displays a grid of video thumbnails, including 'Tears in Love - Mashup', 'Rangabati - Ram Sampath', 'Salman Khan & Sooraj Barjatya EXCLUSIVE INTERVIEW', 'Bajrangi Bhaijaan | Official Trailer with Subtitles | Salman Khan...', and 'Watch! Salman Khan's 'Sultan' | EID 2016'. The bottom of the page shows a URL: 'https://www.youtube.com/watch?v=uhS7h8hubQ'.

Примеры рекомендаций



Рекомендации — новый поиск

“We are leaving the age of information and entering the age of recommendation”

Chris Anderson, “The Long Tail”

“The Web, they say, is leaving the era of search and entering one of discovery. What's the difference? Search is what you do when you're looking for something. Discovery is when something wonderful that you didn't know existed, or didn't know how to ask for, finds you.”

CNN Money, The race to create a ‘smart’ Google

Информации слишком много



“В 2015 г. дневное потребление информации на человека составит 74 ГБ”

UCSD Study 2014, USA

ECONTENTA

Зачем нужны рекомендации

Для пользователя

- Уменьшение времени на поиск
- Обнаружение нового контента
- Увеличение удовлетворенности от пользования сервисом

Для бизнеса

- Увеличение вовлеченности пользователей
- Увеличение количества пользователей
- Доведение контента только до заинтересованных пользователей
- Увеличение прибыли

Ценность рекомендаций

- Netflix: 2/3 просмотров происходит в результате рекомендаций
- Google News: рекомендации увеличивают CTR на 38%
- Amazon: 35% продаж происходит в результате рекомендаций
- E-Contenta: CTR по рекламным баннерам вырос на 80%

Задача рекомендательной системы

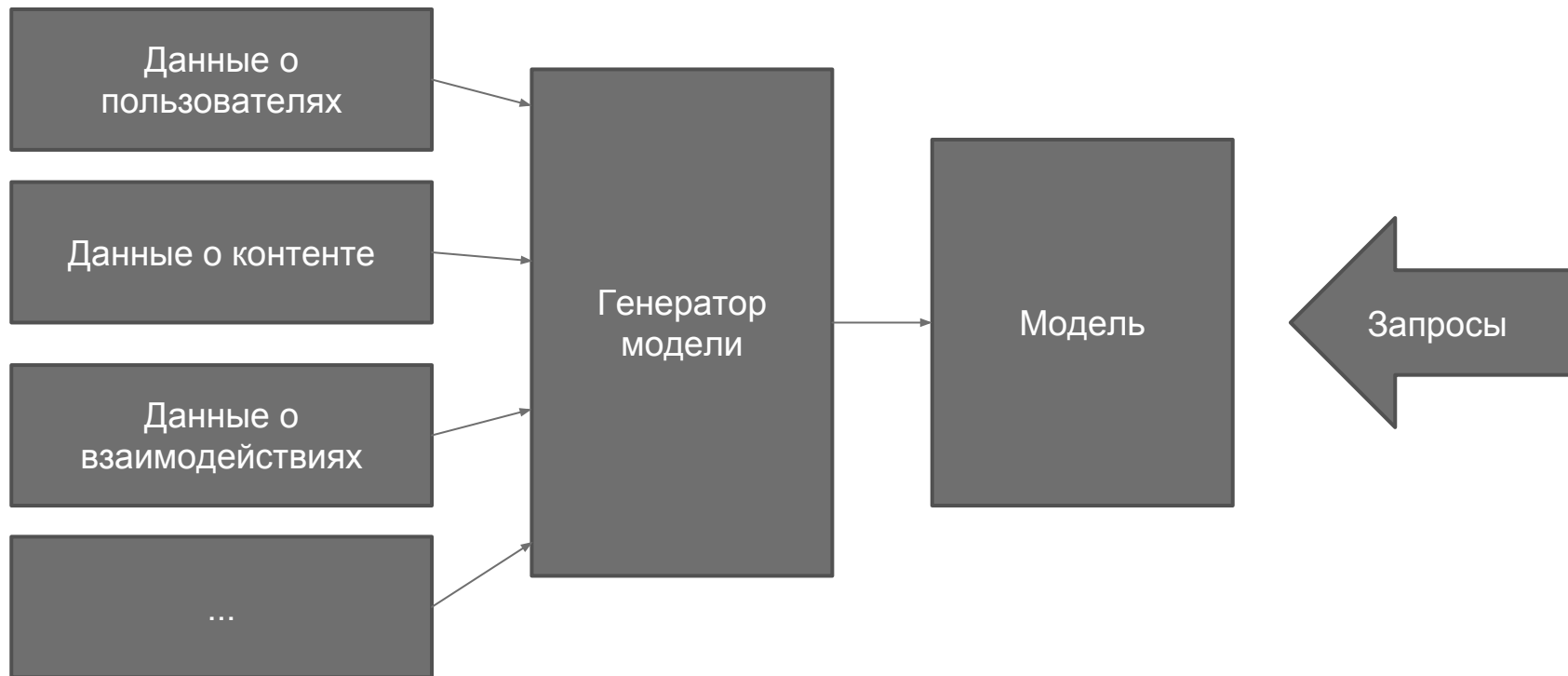
Построить *функцию интересности*, которая способна предсказывать, насколько контент будет интересен пользователю.

На основании данных о контенте, пользователях, их взаимодействиях и окружении.

Задача рекомендательной системы

- \mathbf{U} — множество пользователей, \mathbf{C} — множество контента
- Пусть известна \mathbf{s} — истинная функция интересности
- Пользователю $u \in \mathbf{U}$ рекомендуем $c = \operatorname{argmax}_c(\mathbf{s}(u, c))$
- Задача — построить приближение \mathbf{s}

Схема работы РС



ECONTENTA

Данные для рекомендательных систем

1. Данные о пользователях

- связи между пользователями
- демографические признаки

2. Данные о контенте

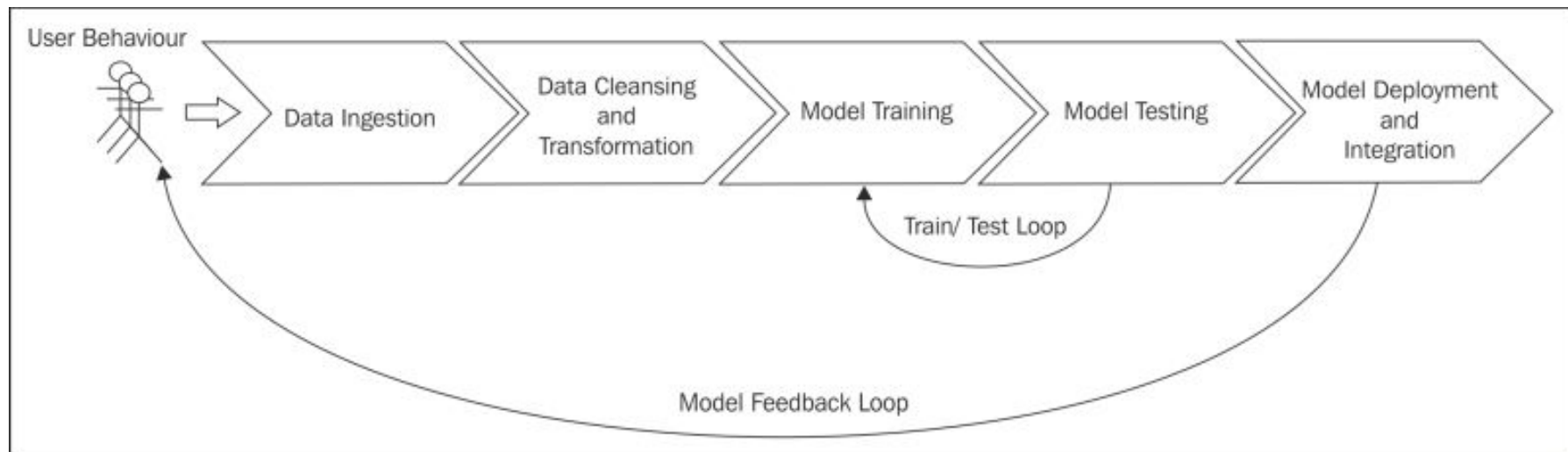
- близость между контентом
- описание контента

3. Данные о взаимодействиях пользователей с контентом

- явный (explicit) отклик
- неявный (implicit) отклик

4. Данные о контексте

РС — область применения ML



РС — не только ML

- Интерфейс
- Технические аспекты
 - Масштабируемость
 - Скорость работы
 - Эффективность использования ресурсов
- Этические аспекты
- Трудно формализуемая метрика качества

Виды алгоритмов рекомендаций

1. Коллаборативная фильтрация
2. Контентные рекомендации
3. Демографические рекомендации
4. Социальные рекомендации
5. Ранжирование
6. Гибридные рекомендации

Пример. Рекомендация фильмов

User\Film	Матрица	Эквилибриум	Гордость и предубеждение	Разум и чувства
Оля	5		10	9
Тимур	8	10		
Лёва	9	8	5	4
Агата		2	8	7
Кеша	8	7	8	8

1. Как предсказать рейтинг?

2. Какие есть сложности?

2. Классические методы

2.1. Контентные рекомендации

Описание контента

- Явные атрибуты контента: год производства, жанры, продолжительность, цена, ...
- Текстовый контент: название, описание, содержимое, ...
- Сложный контент: звук, изображения, видео.

Требуют преобразования в численные характеристики.

Контентные рекомендации

- Основаны на описании контента, а не о информации о пользователях/их взаимодействиях
- Контентная РС рекомендует контент, похожий на тот, который понравился пользователю в прошлом
- Составляет профиль пользователя на предыдущего контента, с которым он взаимодействовал в прошлом.
- Выдает рекомендации на основе близости контента к составленному профилю

Контентные рекомендации

- Контент представлен в виде набора признаков
- На основе истории взаимодействий пользователя строится *модель предпочтений пользователя*
 - Средние значения признаков понравившегося контента
 - Классификатор, обученный на векторах признаков контента
- Интересность нового контента оценивается моделью предпочтений пользователя на основе его признаков

Контентный подход. Пример 1

1. В нашем распоряжении информация о фильмах: название и жанры
2. Пользователю понравились: Star Wars и Star Troopers
3. Вывод: пользователю нравятся фильмы :
 - жанра “фантастика” и “приключения”
 - содержащие “star” в названии
4. Выберем все фильмы удовлетворяющим этим условиям с учетом популярности
 - Результат: Star Trek

Контентный подход. Пример 2

1. Описание контента => вектор:

а. жанр “фантастика” и “приключения” =>

1	0	0	1	0	0
---	---	---	---	---	---

б. рейтинг на КиноПоиске 8 =>

8

2. Результат: вектор контента $v_j =$

1	0	0	1	0	0	8
---	---	---	---	---	---	---

3. Средний вектор предпочтений по всем понравившимся

$$u = \frac{1}{|I(u)|} \sum_{j \in I(u)} v_j$$

Рекомендуем контент расстояние от вектора которого до u_i минимально

$$j_{rec} = \arg \max_j \text{dist}(u, v_j)$$

Контентный подход: pros и cons

Pros:

- + Нет необходимости в других пользователях
- + Работает в случае уникальных вкусов
- + Может рекомендовать новый и непопулярный контент
- + Интерпретируем (в зависимости от сложности модели предпочтений пользователей)

Cons:

- Выделение признаков — трудоемкий процесс (видео, аудио)
- Признаки могут не содержать информацию, мотивирующую пользователя к взаимодействию
- Качество рекомендаций не растет с ростом данных
- Эффект пузыря

2.1. Коллаборативная фильтрация

Коллаборативная фильтрация

1. Коллаборативная фильтрация на основе пользователей (user-based collaborative filtering)
 - a. Найти пользователей с похожими вкусами
 - b. Рекомендовать то, что им нравится

2. Коллаборативная фильтрация на основе контента (item-based collaborative filtering)
 - a. Две единицы контента похожи, если с ними взаимодействовали одни и те же пользователи
 - b. Рекомендовать контент, похожий на тот, с которым пользователь взаимодействовал ранее

КФ на основе пользователя

1. Пользователи u_j , $j=1..n$ и контент c_j , $j=1..m$.
2. V — матрица рейтингов n на m : $v_{i,j}$ — рейтинг i -го пользователя j -му контенту
3. Близость между пользователями оценивается по формуле:

$$u_{ik} = \frac{\sum_j (v_{ij} - v_i)(v_{kj} - v_k)}{\sqrt{\sum_j (v_{ij} - v_i)^2 \sum_j (v_{kj} - v_k)^2}} \quad \text{или} \quad \cos(u_i, u_j) = \frac{\sum_{k=1}^m v_{ik} v_{jk}}{\sqrt{\sum_{k=1}^m v_{ik}^2 \sum_{k=1}^m v_{jk}^2}}$$











4. Рейтинг оценивается по формуле:

$$v_{ij}^* = K \sum_{v_{kj} \neq ?} u_{jk} v_{kj} \quad \text{или} \quad v_{ij}^* = v_i + K \sum_{v_{kj} \neq ?} u_{jk} (v_{kj} - v_k)$$





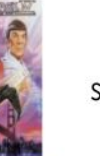



КФ на основе пользователя. Пример

							$\text{sim}(u,v)$
	2			4	5		NA
	5		4			1	
			5		2		
		1		5		4	
			4			2	
	4	5		1			NA









КФ на основе пользователя. Пример

							$\text{sim}(u,v)$
	2			4	5		NA
	5		4			1	0.87
			5		2		
		1		5		4	
			4			2	
	4	5		1			NA











КФ на основе пользователя. Пример

							$\text{sim}(u,v)$
	2			4	5		NA
	5		4			1	0.87
			5		2		1
		1		5		4	
			4			2	
	4	5		1			NA

КФ на основе пользователя. Пример

							$\text{sim}(u,v)$
	2			4	5		NA
	5		4			1	0.87
			5		2		1
		1		5		4	-1
			4			2	
	4	5		1			NA

КФ на основе пользователя. Пример

							$\text{sim}(u,v)$
	2			4	5		NA
	5		4			1	0.87
			5		2		1
		1		5		4	-1
	3.51*	3.81*	4	2.42*	2.48*	2	
	4	5		1			NA

КФ на основе контента

1. Выбрать пользователя и контент для оценки рейтинга
2. Рассчитать близость между целевым контентом и контентом пользователя
3. Выбрать K ближайших единиц контента
4. Рассчитать взвешенное среднее рейтингов пользователя по K единицам контента

$$P_{u,i} = \frac{\sum_{all\ similar\ items, N} (S_{i,N} * R_{u,N})}{\sum_{all\ similar\ items, N} (|S_{i,N}|)}$$

КФ на основе контента. Меры близости

- Косинусова близость

$$S(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|_2 * \|\vec{j}\|_2}$$


- Корреляция






$$S(i, j) = \text{corr}_{\vec{i}, \vec{j}} = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_i)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_j)^2}}$$

- Нормированная косинусова близость

$$S(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}}$$

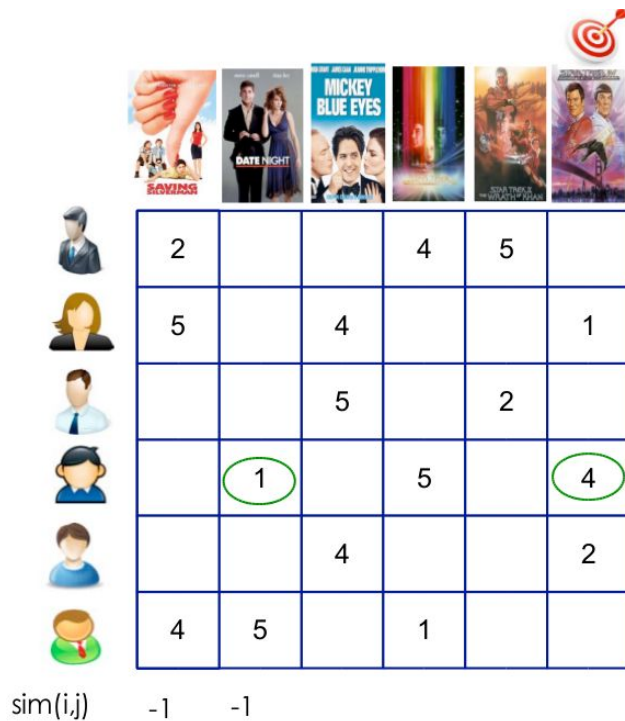
КФ на основе контента. Пример



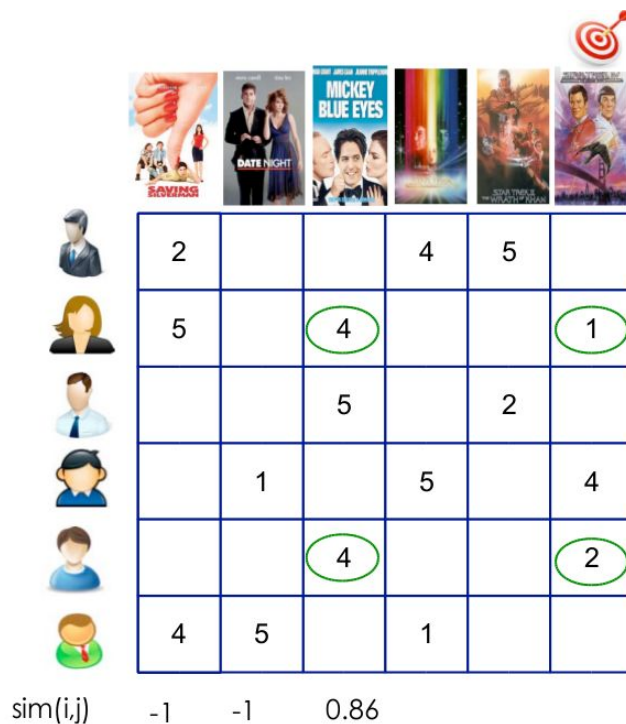
	2			4	5	
	5		4			1
			5		2	
		1		5		4
			4			2
	4	5		1		

sim(i,j) -1


КФ на основе контента. Пример









КФ на основе контента. Пример





КФ на основе контента. Пример



	2			4	5	
	5		4			1
			5		2	
		1		5		4
			4			2
	4	5		1		
$\text{sim}(i,j)$	-1	-1	0.86	1		

КФ на основе контента. Пример












	2			4	5	
	5		4			1
			5		2	
		1		5		4
			4			2
	4	5		1		
$\text{sim}(i,j)$	-1	-1	0.86	1	NA	

$\text{sim}(6,5)$ cannot
be calculated



КФ на основе контента. Пример

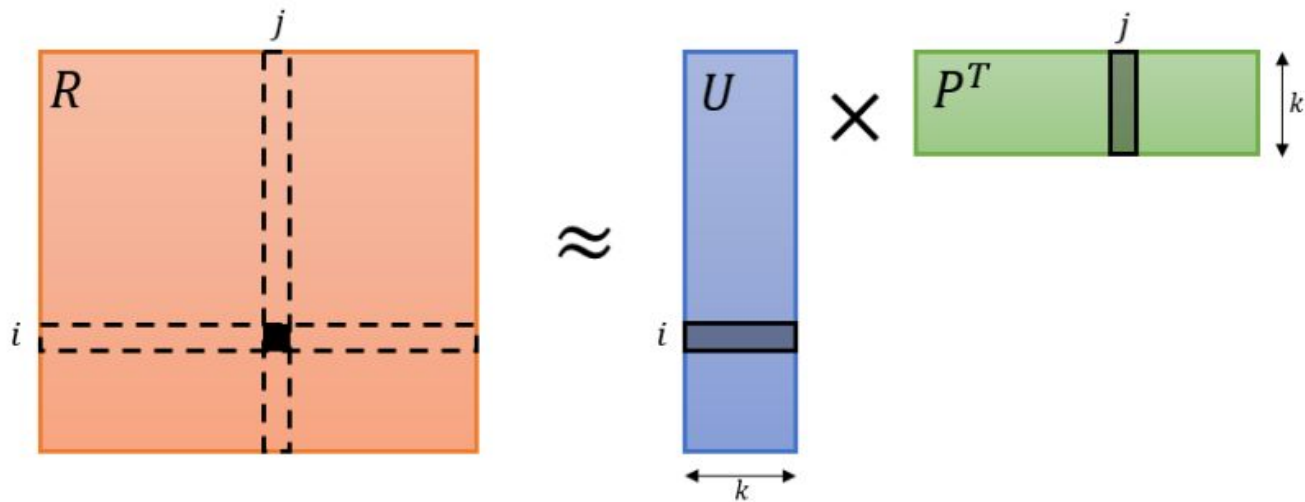
	2			4	5	2.94*
	5		4			1
			5		2	2.48*
		1		5		4
			4			2
	4	5		1		1.12*
sim(i,j)	-1	-1	0.86	1	NA	

КФ на основе пользователя/контента. Сложности

- Разреженность — пользователь взаимодействует с $< 1\%$ контента, с большей частью контента взаимодействует $< 1\%$ пользователей
 - Снижение размерности
- Масштабируемость — сложность увеличивается с ростом кол-ва пользователей и контента.
 - В КФ на основе контента похожесть контента можно рассчитать заранее
- Холодный старт
 - Пользователя
 - Контента
 - Системы

Матричная факторизация

Матричная факторизация



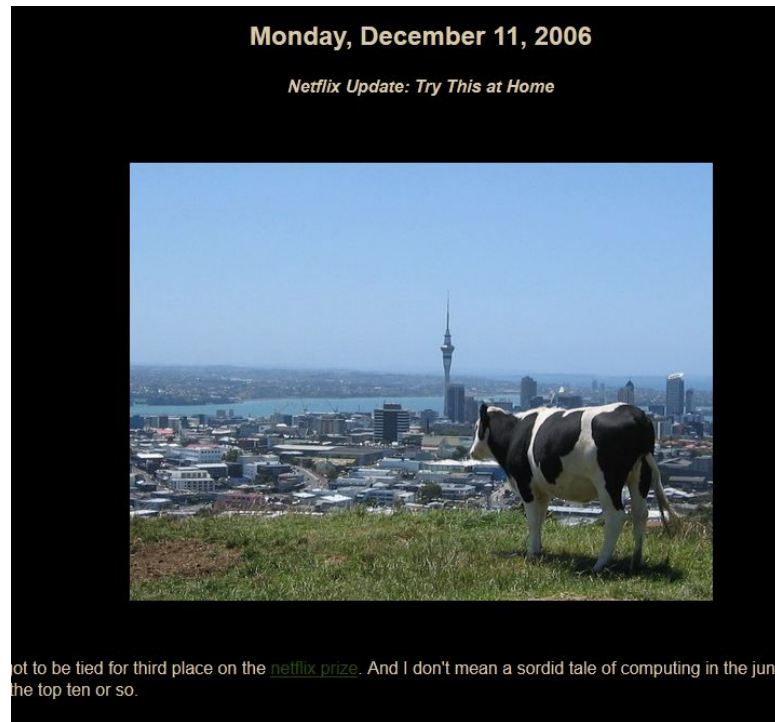
SVD

$$R = U S V^T$$

- R — матрица взаимодействий пользователей с контентом
- U — матрица левых с.в.
- S — матрица с.ч. (“сила факторов”)
- V — матрица правых с.в.
- $US^{1/2}$ — факторы пользователей
- $S^{1/2}V$ — факторы пользователей

Simon Funk's SVD

- Инкрементальный приближенный расчет SVD с помощью градиентного спуска
- Предложен в рамках Netflix Prize, на момент публикации занял третье место



SVD++

- Бейзлайн: $b_{uv} = \mu + b_u + b_v$
- Вектор факторов пользователя $p_u \in \mathbb{R}^f$ и контента $q_v \in \mathbb{R}^f$
- Предсказание рейтинга: $r'_{uv} = b_{uv} + p_u^T q_v$

- SVD++:
$$r_{ui} = \mu + b_u + b_i + \left(p_u + \sum_{i \in Ufeed(u)} \alpha_i y_i \right)^T q_i.$$

- $Ufeed(u)$ — взаимодействия пользователя

- $\alpha_i = \frac{1}{\sqrt{|N(u)|}}$ либо $\frac{r_{uj} - b_u}{\sqrt{|R(u)|}}$

Контентный VS коллаборативный подходы

Контентный

- + не требователен к истории взаимодействий
- + может рекомендовать новый и непопулярный контент
- + может соответствовать узким вкусам
- + легко интерпретируем
- создание вектора признаков — сложная задача
- зачастую невысокое качество рекомендаций
- не учитывает контекст

Коллаборативный

- + не требует наличия данных о контенте
- + в большинстве случаев демонстрирует лучшее качество
- + не зависит от предметной области
- требователен к уникальности контента
- качество сильно зависит от полноты истории
- сдвиг в сторону популярности
- проблема холодного старта
- не учитывает контекст

Контентный VS коллаборативный подходы

Recommending New Movies: Even a Few Ratings Are More Valuable Than Metadata

István Pilászy *
Dept. of Measurement and Information Systems
Budapest University of Technology and
Economics
Magyar Tudósok krt. 2.
Budapest, Hungary
pila@mit.bme.hu

Domonkos Tikk *†
Dept. of Telecom. and Media Informatics
Budapest University of Technology and
Economics
Magyar Tudósok krt. 2.
Budapest, Hungary
tikk@tmit.bme.hu

ABSTRACT

The Netflix Prize (NP) competition gave much attention to collaborative filtering (CF) approaches. Matrix factorization (MF) based CF approaches assign low dimensional feature vectors to users and items. We link CF and content-based filtering (CBF) by finding a linear transformation that transforms user or item descriptions so that they are as close as possible to the feature vectors generated by MF for CF.

We propose methods for explicit feedback that are able to handle 140 000 features when feature vectors are very sparse. With movie metadata collected for the NP movies we show that the prediction performance of the methods is comparable to that of CF, and can be used to predict user preferences on new movies.

We also investigate the value of movie metadata compared to movie ratings in regards of predictive power. We compare

1. INTRODUCTION

The goal of recommender systems is to give personalized recommendation on items to users. Typically the recommendation is based on the former and current activity of the users, and metadata about users and items, if available.

There are two basic strategies that can be applied when generating recommendations. Collaborative filtering (CF) methods are based only on the activity of users, while content-based filtering (CBF) methods use only metadata. In this paper we propose hybrid methods, which try to benefit from both information sources.

The two most important families of CF methods are matrix factorization (MF) and neighbor-based approaches. Usually, the goal of MF is to find a low dimensional representation for both users and movies, i.e. each user and movie is associated with a feature vector. Movie metadata (which can mostly textual) can also be represented as a vector using

Другие подходы

Кластеризация

1. Пользователи кластеризуются в пространстве контента/признаков
 2. Каждый пользователь относится к кластеру
 3. В рамках кластера формируется единый список рекомендаций
 4. Рекомендации пользователю — рекомендации его кластера
-
- + Учитывает скрытые связи между пользователем и контентом
 - + Может быть использован для уменьшения кол-ва контента
 - Рекомендации по кластеру менее релевантны, чем персональные

Ассоциативные правила

- Пары/тройки/... контента, с которыми часто взаимодействуют одновременно

- + Есть быстрые, эффективные реализации
- Может выдавать некачественные рекомендации

		Book1	Book2	Book3	Book4	Book5	Book6
	Customer A	X			X		
	Customer B		X	X		X	
	Customer C		X	X			
	Customer D		X				X
	Customer E	X				X	
	Customer F			X		X	

		Book1	Book2	Book3	Book4	Book5	Book6
Customers who bought...	Book1				1	1	
	Book2			2		1	1
	Book3		2			2	
	Book4	1					
	Book5	1	1	2			
	Book6		1				

ECONTENTA

3. Современные методы

Как оценивать качество?

Математика:

- Метрики регрессии: MAE, RMSE, R-squared
- Метрики классификации: accuracy, log loss, roc auc

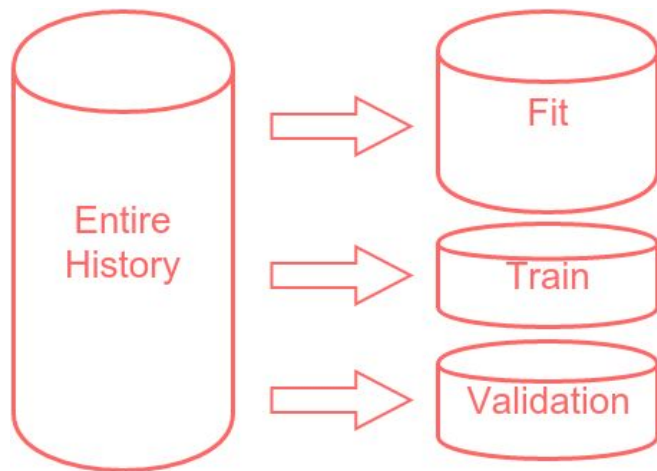
Бизнес:

- Конверсии
- Деньги

Рекомендации как классификация

- Каждое взаимодействие — наблюдение с бинарной меткой
 - like / dislike
 - $\text{rating} > 3$ / $\text{rating} \leq 3$
 - clicked / not clicked
- Разные типы действий
 - Вес
 - Многоклассовая классификация
 - Многоклассовая пересекающаяся классификация
- Необходимо сформировать вектор признаков

Обучающая выборка



Вектор признаков



Рекомендации как классификация

- + Универсальность
 - + Разные данные — один подход
- + Гибкость
 - + Логистическая регрессия, SVM, деревья решений, градиентный бустинг, ...
- Выделение признаков — трудоемкая задача
- Сформировать тренировочную выборку бывает непросто
- Зазор между offline и online
- Риск переобучения

Контекстные рекомендации (context-aware recommendations)

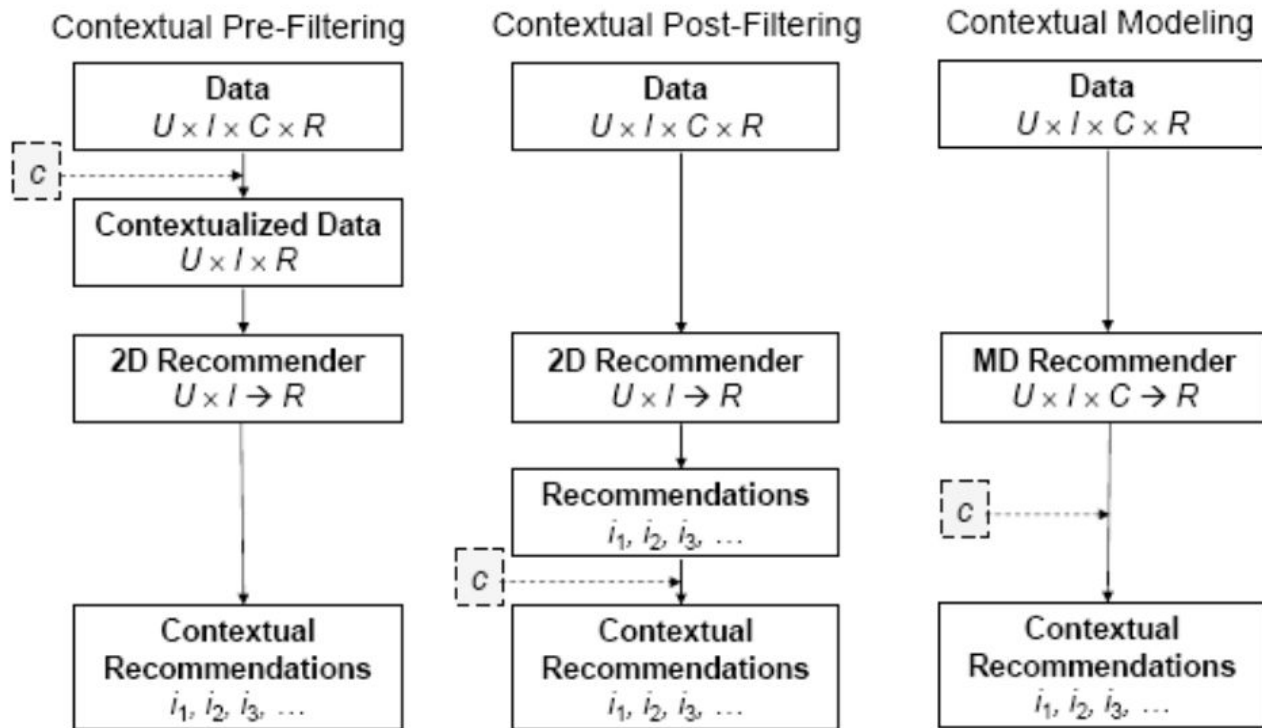
1. Пре-фильтрация
2. Пост-фильтрация
3. Моделирование контекста

Контекст:

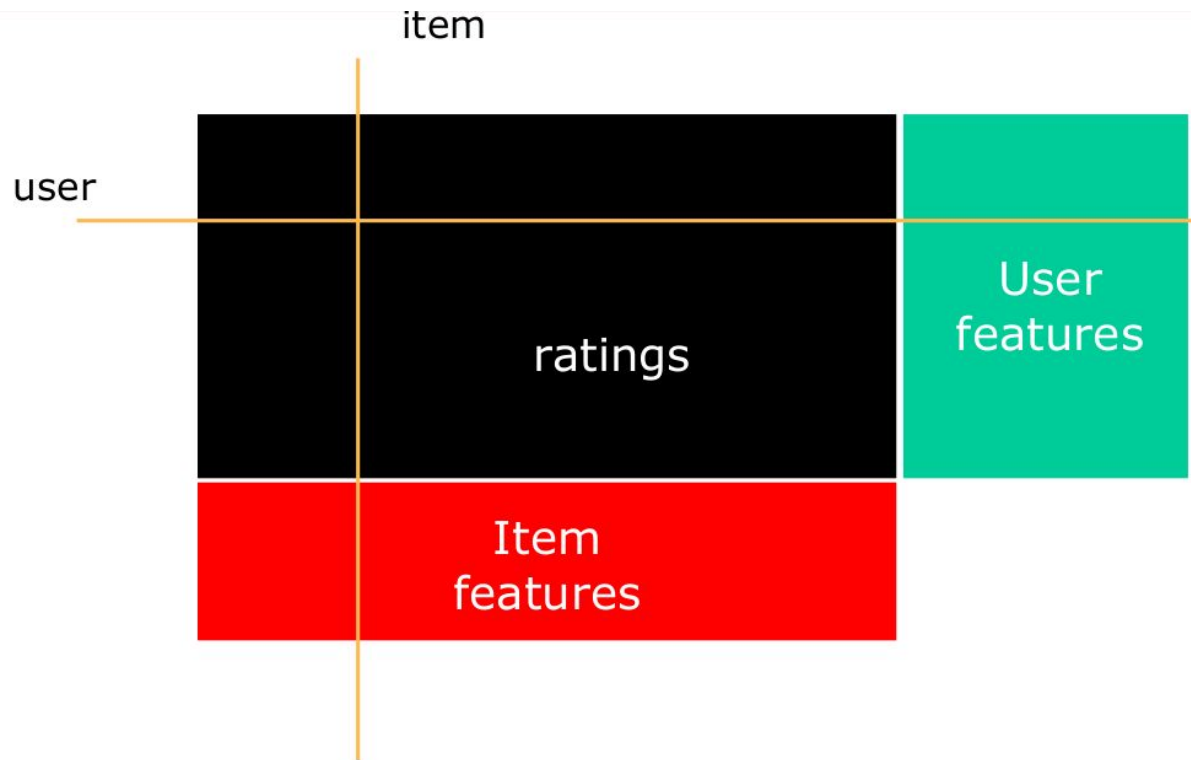
- признаки времени
- признаки окружения
- ...

Контекстные рекомендации

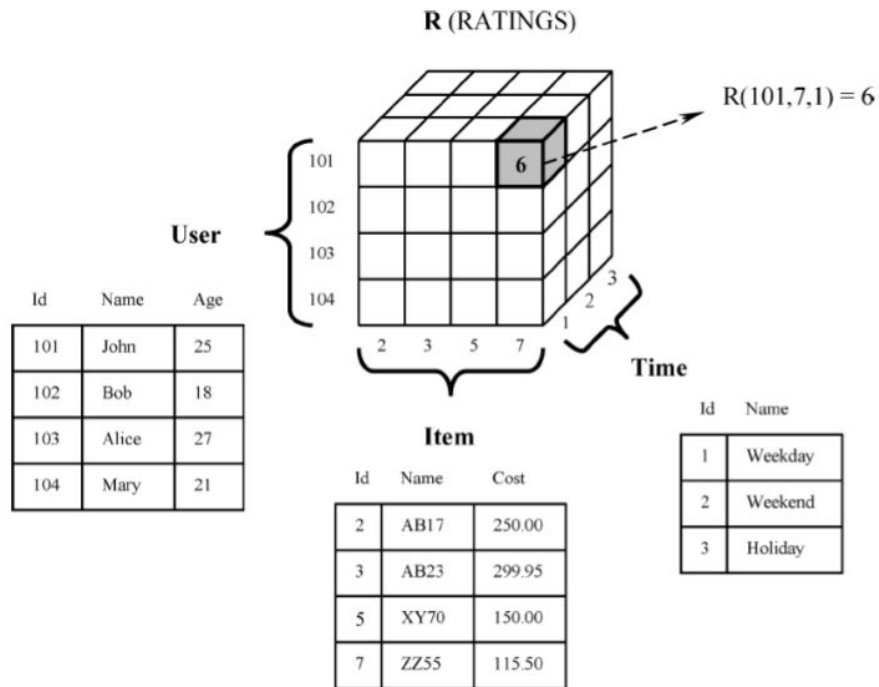
From Adomavicius, Tuzhilin, 2008



2D модель данных

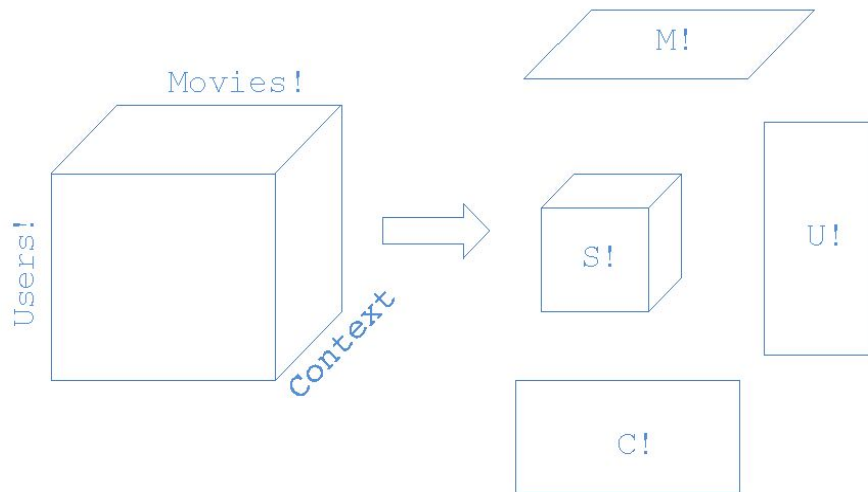


3D модель данных



[Adomavicius et al., 2005]

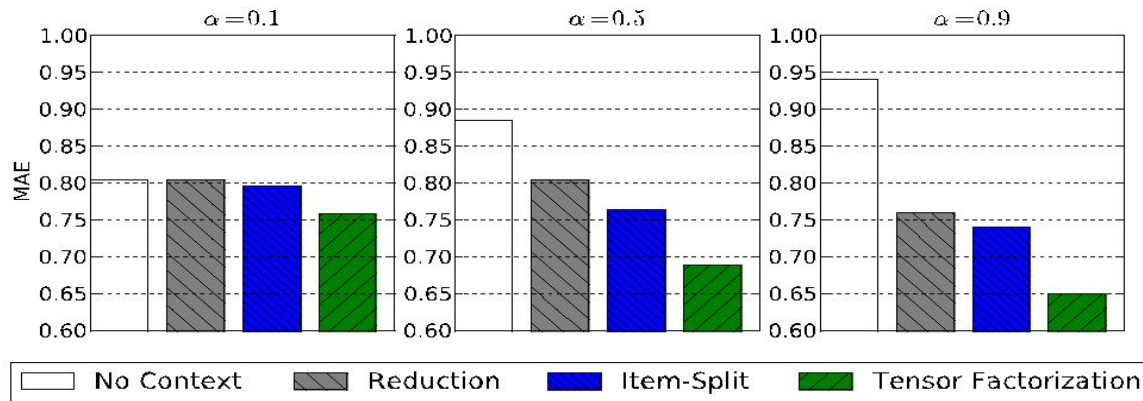
Тензорная факторизация. Модель HOSVD



$$F_{ijk} = S_{ijk} \rightarrow_U U_{ik} \rightarrow_M M_{jk} \rightarrow_C C_{kk}$$

$$R[U, M, C, S] := L(F, Y) + \lambda [U, M, C] + \lambda [S]$$

Тензорная факторизация. Модель HOSVD



Factorization Machines

- Объединение регрессионного подхода и подхода тензорной факторизации
- Проблемы тензорного разложения
 - Каждый новый вариант матричного/тензорного разложения требует нового алгоритма обучения
 - Сложно учитывать вещественные признаки
 - Сложно добавлять новые типы данных

Factorization Machines

- Вход модели — вектор:
 - вектор пользователя + вектор контента + вектор контекста +=> один большой вектор
- = 
- Классические методы ML не справляются
 - линейная регрессия не учитывает взаимодействий между признаками
 - квадратичная требует $d*d$ памяти
 - Каждому признаку по вектору в пространстве скрытых факторов
 - Взаимодействие между признаками — произведение векторов факторов

Factorization Machines

- Параметры

$$b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d, \mathbf{V} \in \mathbb{R}^{d \times k}$$

- Модель

$$\begin{aligned} f(\mathbf{x}) &= b + \sum_{i=1}^d w_i x_i + \sum_{i=1}^d \sum_{j=i+1}^d x_i x_j \mathbf{v}_i^T \mathbf{v}_j && \mathcal{O}(d^2) \\ &= b + \sum_{i=1}^d w_i x_i + \frac{1}{2} \sum_{f=1}^k \left(\left(\sum_{i=1}^d x_i v_{i,f} \right)^2 - \sum_{i=1}^d x_i^2 v_{i,f}^2 \right) && \mathcal{O}(dk) \end{aligned}$$

Factorization Machines. Пример

- Вектор: пользователь + контент

Feature vector \mathbf{x}										
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	
	A	B	C	...	TI	NH	SW	ST	...	
	User				Movie					

- FM — аналог MF

$$f(\mathbf{x}) = b + w_u + w_i + \mathbf{v}_u^T \mathbf{v}_i$$

Factorization Machines. Пример

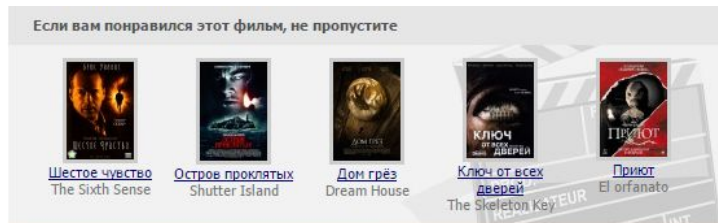
- Вектор: пользователь + контент + время

Feature vector \mathbf{x}											
$\mathbf{x}^{(1)}$	1	0	0	...	1	0	0	0	...	0.2	
$\mathbf{x}^{(2)}$	1	0	0	...	0	1	0	0	...	0.6	
$\mathbf{x}^{(3)}$	1	0	0	...	0	0	1	0	...	0.61	
$\mathbf{x}^{(4)}$	0	1	0	...	0	0	1	0	...	0.3	
$\mathbf{x}^{(5)}$	0	1	0	...	0	0	0	1	...	0.5	
$\mathbf{x}^{(6)}$	0	0	1	...	1	0	0	0	...	0.1	
$\mathbf{x}^{(7)}$	0	0	1	...	0	0	1	0	...	0.8	
	A	B	C	...	TI	NH	SW	ST	...		
	User				Movie					Time	

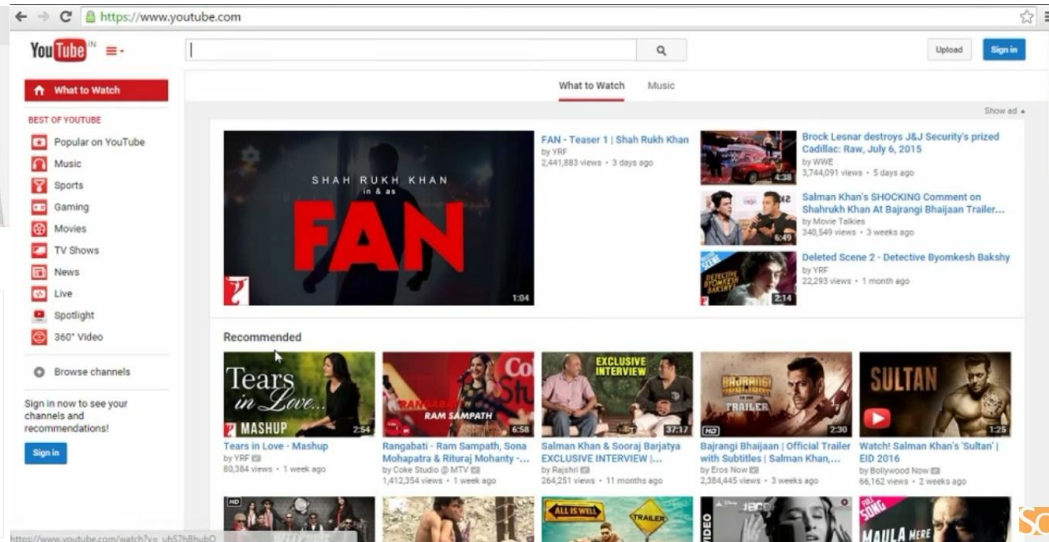
- FM

$$f(\mathbf{x}) = b + w_u + w_i + x_t w_t + \mathbf{v}_u^T \mathbf{v}_i + x_t \mathbf{v}_u^T \mathbf{v}_t + x_t \mathbf{v}_i^T \mathbf{v}_t$$

Рекомендации как задача ранжирования



Ищут вместе с исполнителями, которых вы слушаете



Рекомендации как задача ранжирования

- Рекомендации — упорядоченный список
- Различие в рейтинге неважно
- Важен порядок в топе
- Метрики качества регрессии/классификации не годятся

Метрики качества ранжирования

- Метрики классификации/регрессии не подходят для ранжирования
- Метрике качества ранжирования следует учитывать отличия в верхней части списка сильнее, чем в нижней.



Метрики качества ранжирования. $map@K$

- Precision at K

$$p@K = \frac{\sum_{k=1}^K r^{true}(\pi^{-1}(k))}{K} = \frac{\text{релевантных элементов}}{K}.$$

- Average precision at K

$$ap@K = \frac{1}{K} \sum_{k=1}^K r^{true}(\pi^{-1}(k)) \cdot p@k.$$

- Mean average precision at K

$$map@K = \frac{1}{N} \sum_{j=1}^N ap@K_j.$$

Метрики качества ранжирования. NDCG@K

- Cumulative gain at K

$$CG@K = \sum_{k=1}^K r^{true}(\pi^{-1}(k)).$$

- Discounted Cumulative Gain at K

$$DCG@K = \sum_{k=1}^K \frac{2^{r^{true}(\pi^{-1}(k))} - 1}{\log_2(k + 1)}.$$

- Normalized Discounted Cumulative Gain at K

$$nDCG@K = \frac{DCG@K}{IDCG@K},$$

Метрики качества ранжирования. $MRR@K$

- Reciprocal Rank at K

$$RR@K = \frac{1}{\min\{k \in [1...K] : r^{true}(\pi^{-1}(k)) = 1\}},$$

- Mean Reciprocal Rank at K

$$MRR@K = \frac{1}{N} \sum_{j=1}^N RR@K_j,$$

Метрики качества ранжирования

- Метрики на основе ранговой корреляции
 - Ранговый коэффициент корреляции Спирмена
 - Ранговый коэффициент корреляции Кендэлла
- Метрики на основе каскадной модели поведения
 - Expected Reciprocal Rank
 - PFound

Подходы к ранжированию. Pointwise & Pairwise

1. Pointwise

- a. Сводится к задаче классификации/регрессии
- b. Рекомендации сортируются по предсказанному значению

2. Pairwise

- a. Наблюдения рассматриваются попарно
 $\{(x_k, y_k)\}_k \Rightarrow \{((x_k, x_j), y_k > y_j)\}_{k,j}$
- b. Цель — минимизировать число перестановок
- c. Решается как задача классификации

Подходы к ранжированию. Listwise

- Наблюдение — упорядоченный список: $\{ ((x_{j1}, \dots, x_{jK}), (y_{j1}, \dots, y_{jK})) \}$
- Косвенные целевые функции:
 - RankCosine — косинусова близость между истинным и полученным списком
 - ListNet — расстояние Кульбака-Лейблера
 - Могут не коррелировать с метриками ранжирования
- Непосредственная минимизация метрик ранжирования
 - Методы оптимизации нулевого порядка (без градиента)
 - Генетическое программирование
 - Алгоритм имитации отжига
 - Аппроксимация метрик ранжирования
 - SVM-MAP, TFMAP, CLiMF
 - Бустинг (AdaRank)

4. Заключение

Виды гибридизации

1. Линейная комбинация
2. Контекстное переключение
3. Смешивание
4. Комбинирование признаков
5. Каскадная
6. Стэкинг

Советы

- Важно правильно поставить задачу
- Построить методику оценки
- Всегда начинать с простого, двигаться итеративно
- Программирование >> машинное обучение

Спасибо!

5. Практика

Задача

Построить алгоритм рекомендаций на данных MovieLens-10M

1. Скачать данные (ratings.dat & movies.dat), разобраться с форматом, прочитать
2. Реализовать метрику качества ранжирования map@10
3. Разбить данные на train, validation, test по времени
4. Реализовать baseline: самое популярные.
5. Улучшить baseline
 - a. Content based recommendations
 - b. SVD, MF, FM
 - c. Classification/Regression
 - d. Ranking (xgboost(objective="rank:pairwise"))