

Коллаборативная фильтрация

К. В. Воронцов
vokov@forecsys.ru

Этот курс доступен на странице вики-ресурса
<http://www.MachineLearning.ru/wiki>
«Машинное обучение (курс лекций, К.В.Воронцов)»

ноябрь 2012

Содержание

- 1 Постановка задачи и приложения**
 - Постановка задачи
 - Примеры приложений
 - Модели коллаборативной фильтрации
- 2 Корреляционные модели**
 - Модели, основанные на хранении данных
 - Задача восстановления пропущенных значений
 - Функции близости
- 3 Латентные модели**
 - Бикластеризация и матричные разложения
 - Вероятностные латентные модели
 - Эксперименты на данных Яндекса

Определения и обозначения

U — множество субъектов (клиентов, пользователей: users);

R — множество объектов (ресурсов, товаров, предметов: items);

Y — пространство описаний транзакций;

$D = (u_i, r_i, y_i)_{i=1}^m \in U \times R \times Y$ — транзакционные данные;

Агрегированные данные:

$F = \|f_{ur}\|$ — матрица кросс-табуляции размера $|U| \times |R|$,

где $f_{ur} = \text{aggr}\{(u_i, r_i, y_i) \in D \mid u_i = u, r_i = r\}$

Задачи:

- прогнозирование незаполненных ячеек f_{ur} ;
- оценивание сходства: $\rho(u, u')$, $\rho(r, r')$, $\rho(u, r)$;
- одновременная кластеризация множеств U и R ;
- выявление латентных интересов $p(t|u)$, $q(t|r)$ относительно заданного либо неизвестного набора тем $t = 1, \dots, T$.

Пример 1. Рекомендательная система по посещениям

U — пользователи Интернет;

R — ресурсы (сайты, страницы, документы, новости, и т.п.);

f_{ur} = [пользователь u посетил ресурс r];

Основная гипотеза Web Usage Mining:

- Действия (посещения) пользователя характеризуют его интересы, вкусы, привычки, возможности.

Задачи персонализации предложений:

- для пользователя u :
 - выдать оценку ресурса r ;
 - выдать ранжированный список рекомендуемых ресурсов;
- для ресурса r : выдать список ресурсов, близких к r .

Пример реализации: <http://SurfingBird.ru/>

Пример 2. Рекомендательная система по покупкам

U — клиенты интернет-магазина (amazon.com и др.);

R — товары (книги, видео, музыка, и т.п.);

f_{ur} = [клиент u купил товар r];

Задачи персонализации предложений:

- выдать оценку товара r для клиента u ;
- выдать клиенту u список рекомендуемых товаров;
- предложить скидку на совместную покупку (cross-selling);
- информировать клиента о новом товаре (up-selling);
- сегментировать клиентскую базу;
выделить интересы клиентов (найти целевые аудитории).

Пример 3. Рекомендательная система на основе рейтингов

U — клиенты интернет-магазина (netflix.com и др.);

R — товары (книги, видео, музыка, и т.п.);

f_{ur} = рейтинг, который клиент u выставил товару r ;

Задачи персонализации предложений — те же.

Пример: конкурс Netflix [www.netflixprize.com]

- 2 октября 2006 — 21 сентября 2009; главный приз — \$10⁶;
- $|U| = 0.48 \cdot 10^6$; $|R| = 17 \cdot 10^3$;
- 10⁸ рейтингов $\{1, 2, 3, 4, 5\}$;
- точность прогнозов оценивается по тестовой выборке D' :

$$\text{RMSE}^2 = \frac{1}{|D'|} \sum_{(u,r) \in D'} (f_{ur} - \hat{f}_{ur})^2;$$

- задача: уменьшить RMSE с 0.9514 до 0.8563 (на 10%).

Пример 4. Анализ текстов

U — текстовые документы (статьи, новости, и т.п.);

R — ключевые слова или выражения;

f_{ur} = частота встречаемости слова r в тексте u .

Задачи тематического моделирования (topic modeling):

- кластеризовать тексты по темам;
определить число тем (кластеров) в коллекции;
- построить иерархический каталог тем;
определить, на какие подтемы разбивается каждая тема;
- по тексту u найти тексты близкой тематики;
- проследить динамику развития темы во времени;
- найти публикации, авторов, организации, журналы, конференции по теме.

Пример 5. Анализ изображений

U — изображения;

R — найденные на изображениях элементы;

$f_{ur} = [\text{изображение } u \text{ содержит элемент } r]$.

Задачи тематического моделирования (topic modeling):

- кластеризовать изображения по темам;
- построить иерархический каталог тем;
- по изображению u найти схожие;

Задачи связывания изображений и текстов:

- аннотировать (тегировать) изображение;
- по изображению найти описание;
- по описанию найти изображение;

Пример 6. Социальные сети, форумы, блоги

U — пользователи;

D — текстовые документы (обсуждения, блоги);

W — ключи (ключевые слова или выражения);

f_{ud} = [пользователь u прочитал/написал d];

g_{rw} = частота встречаемости ключа w в тексте d ;

h_{uv} = [пользователь u — друг пользователя v].

Некоторые задачи анализа социальной сети:

- рекомендовать пользователю интересные ему блоги;
- найти единомышленников (like-minded people);
- описать интересы пользователя ключами;
- найти все блоги, похожие на данный;
- построить иерархический тематический каталог блогов.

Два основных подхода в коллаборативной фильтрации

1 Корреляционные модели

(Memory-Based Collaborative Filtering)

- хранение всей исходной матрицы данных F ;
- сходство клиентов — это корреляция строк матрицы F ;
- сходство объектов — это корреляция столбцов матрицы F .

2 Латентные модели

(Latent Models for Collaborative Filtering)

- оценивание профилей клиентов и объектов
(*профиль — это вектор скрытых характеристик*);
- хранение профилей вместо хранения F ;
- сходство клиентов и объектов — это сходство их профилей.

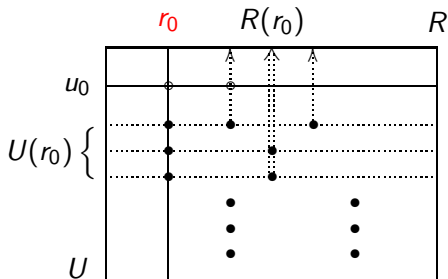
Подборки статей по коллаборативной фильтрации:

jamesthornton.com/cf

ict.ewi.tudelft.nl/~jun/CollaborativeFiltering.html

Тривиальная рекомендательная система

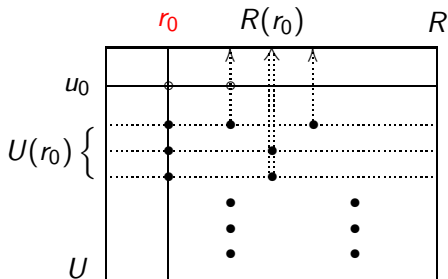
«клиенты, купившие r_0 ,
 также покупали $R(r_0)$ »
 [Amazon.com]



- 1 $U(r_0) := \{u \in U \mid f_{ur_0} \neq \emptyset, u \neq u_0\}$ — коллаборация;
- 2 $R(r_0) := \left\{ r \in R \mid B(r) = \frac{|U(r_0) \cap U(r)|}{|U(r_0) \cup U(r)|} > 0 \right\}$,
 где $B(r)$ — одна из возможных мер близости r к r_0 ;
- 3 отсортировать $R(r_0)$ по убыванию $B(r)$, взять top N .

Тривиальная рекомендательная система

«клиенты, купившие r_0 ,
 также покупали $R(r_0)$ »
 [Amazon.com]

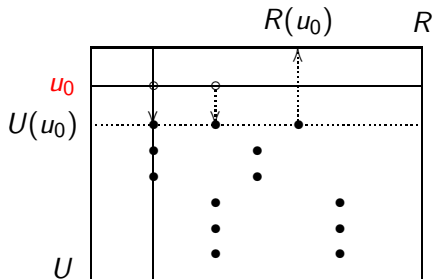


Недостатки:

- рекомендации тривиальны (предлагается всё наиболее популярное);
- не учитываются интересы конкретного пользователя u_0 ;
- проблема «холодного старта»; (новый товар никому не рекомендуется)
- надо хранить всю матрицу F .

От клиента (user-based CF)

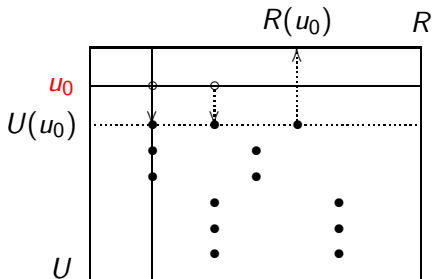
«клиенты, похожие на u_0 ,
 также покупали $R(u_0)$ »



- 1 $U(u_0) := \{u \in U \mid \text{corr}(u_0, u) > \alpha\}$ — коллаборация;
 где $\text{corr}(u_0, u)$ — одна из возможных мер близости u к u_0 ;
- 2 $R(u_0) := \left\{r \in R \mid B(r) = \frac{|U(u_0) \cap U(r)|}{|U(u_0) \cup U(r)|} > 0\right\}$;
 где $U(r) := \{u \in U \mid f_{ur} \neq \emptyset\}$;
- 3 отсортировать $r \in R(u_0)$ по убыванию $B(r)$, взять top N ;

От клиента (user-based CF)

«клиенты, похожие на u_0 ,
также покупали $R(u_0)$ »

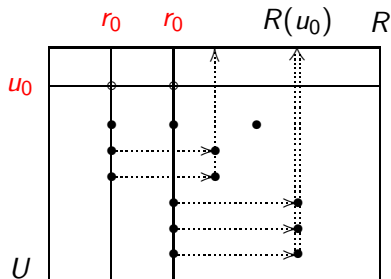


Недостатки:

- рекомендации тривиальны;
- не учитываются интересы конкретного пользователя u_0 ;
- проблема «холодного старта»;
- надо хранить всю матрицу F ;
- **нечего рекомендовать нетипичным/новым пользователям.**

От объекта (item-based CF)

«вместе с объектами,
 которые покупал u_0 ,
 часто покупают $R(u_0)$ »



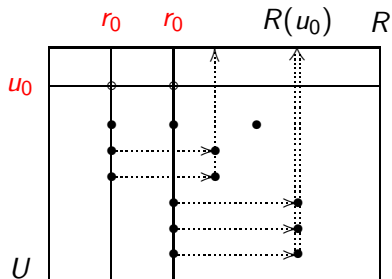
- 1 $R(u_0) := \{r \in R \mid \exists r_0: f_{u_0 r_0} \neq \emptyset \text{ и } B(r) = \text{corr}(r, r_0) > \alpha\}$;
 где $\text{corr}(r, r_0)$ — одна из возможных мер близости r к r_0 ;
- 2 сортировка $r \in R(u_0)$ по убыванию $B(r)$, взять top N ;

От объекта (item-based CF)

«вместе с объектами,
 которые покупал u_0 ,
 часто покупают $R(u_0)$ »

Недостатки:

- рекомендации часто тривиальны (нет коллаборативности);
- проблема «холодного старта»;
- надо хранить всю матрицу F ;
- нечего рекомендовать нетипичным пользователям.



Восстановление пропущенных значений (рейтингов)

Непараметрическая регрессия Надарайя–Ватсона:

$$\hat{f}_{ur} = \bar{f}_u + \frac{\sum_{u' \in U_\alpha(u)} K(u, u')(f_{u'r} - \bar{f}_{u'})}{\sum_{u' \in U_\alpha(u)} K(u, u')},$$

где $\bar{f}_u = \frac{1}{|R(u)|} \sum_{r \in R(u)} f_{ur}$ — средний рейтинг клиента u ,

$R(u)$ — множество объектов, которые клиент u оценил,

$K(u, u')$ — сглаживающее ядро, функция близости u и u' ,

$U_\alpha(u) = \{u' \mid K(u', u) > \alpha\}$ — коллаборация клиента u .

Недостатки:

- проблема «холодного старта»;
- надо хранить всю матрицу F ;

Функции близости, используемые в корреляционных методах

- корреляция Пирсона:

$$K(u, u') = \frac{\sum_{r \in R(u, u')} (f_{ur} - \bar{f}_u)(f_{u'r} - \bar{f}_{u'})}{\sqrt{\sum_{r \in R(u, u')} (f_{ur} - \bar{f}_u)^2 \sum_{r \in R(u, u')} (f_{u'r} - \bar{f}_{u'})^2}};$$

- косинусная мера близости:

$$K(u, u') = \frac{\sum_{r \in R(u, u')} f_{ur} f_{u'r}}{\sqrt{\sum_{r \in R(u, u')} f_{ur}^2 \sum_{r \in R(u, u')} f_{u'r}^2}};$$

где $R(u, u') = \begin{cases} R(u) \cup R(u'), & \text{для бинарных данных,} \\ R(u) \cap R(u'), & \text{для рейтинговых данных.} \end{cases}$

- статистические критерии:

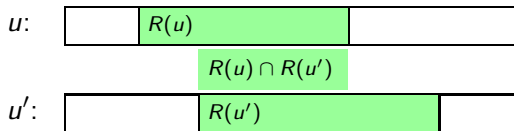
χ^2 , точный тест Фишера (для бинарных данных).

Функции близости на основе точного теста Фишера (FET)

Рассмотрим случай бинарных данных, $f_{ur} \in \{0, 1\}$.

Нулевая гипотеза:

клиенты u и u' совершают свой выбор независимо.



Вероятность случайной реализации r совместных выборов

$$p(r) = P\{|R(u) \cap R(u')| = r\} = \frac{C_{|R(u)|}^r C_{|R|-|R(u)|}^{|R(u')|-r}}{C_{|R|}^{|R(u')|}}.$$

Функция близости $R(u, u') = -\log p(|R(u) \cap R(u')|)$.

Резюме по Memory-Based методам

Преимущества для бизнес-приложений:

- Легко понять.
- Легко реализовать.

Недостатки:

- Не хватает теоретического обоснования:
придумано много способов оценить сходство...
придумано много гибридных (item-user-based) методов...
... и не ясно, что лучше;
- Все методы требуют хранения огромной матрицы F .
- Проблема «холодного старта».

Далее:

- *Латентные модели* — лишены этих недостатков.

Понятие латентной модели

Латентная модель: по данным D оцениваются векторы:

$(p_{tu})_{t \in G}$ — профили клиентов $u \in U$, $|G| \ll |R|$;

$(q_{tr})_{t \in H}$ — профили объектов $r \in R$, $|H| \ll |U|$.

Типы латентных моделей (основные идеи):

1 Ко-кластеризация:

— жёсткая:
$$\begin{cases} p_{tu} = [\text{клиент } u \text{ принадлежит кластеру } t \in G]; \\ q_{tr} = [\text{объект } r \text{ принадлежит кластеру } t \in H]; \end{cases}$$

— мягкая: p_{tu} , q_{tr} — степени принадлежности кластерам.

2 Матричные разложения: $G \equiv H$ — множество тем;
 по p_{tu} , q_{tr} должны восстанавливаться f_{ur} .

3 Вероятностные модели: $G \equiv H$ — множество тем;
 $p_{tu} = p(t|u)$, $q_{tr} = q(t|r)$.

Бикластеризация (ко-кластеризация)

Пусть f_{ur} — вещественные числа или рейтинги;

$g: U \rightarrow G$ — функции кластеризации клиентов ($|G| < \infty$);

$h: R \rightarrow H$ — функции кластеризации объектов ($|H| < \infty$);

Модель усреднения по блокам (Block Average):

$$\hat{f}_{ur}(g, h) = \bar{f}_{g(u),h(r)} + (\bar{f}_u - \bar{f}_{g(u)}) + (\bar{f}_r - \bar{f}_{h(r)});$$

$\bar{f}_{g(u),h(r)}$ — средние по бикластерам;

$\bar{f}_{g(u)}$ и $\bar{f}_{h(r)}$ — средние по кластерам;

\bar{f}_u и \bar{f}_r — средние по клиентам и по объектам;

Функционал качества бикластеризации:

$$\sum_{(u,r) \in D} (\hat{f}_{ur}(g, h) - f_{ur})^2 \rightarrow \min_{g,h};$$

Алгоритм бикластеризации, похожий на k -means

Алгоритм ВВАС (Bregman Block Average Co-clustering)

- 1: инициализировать случайные кластеризации $g(u)$, $h(r)$;
- 2: **пока** кластеризации изменяются
- 3: вычислить средние по бикластерам \bar{f}_{gh} и кластерам \bar{f}_g , \bar{f}_h ;
- 4: вычислить новые кластеризации для всех клиентов $u \in U$:

$$g(u) := \arg \min_g \sum_r (\hat{f}_{ur}(g, h(r)) - f_{ur})^2;$$

- 5: вычислить новые кластеризации для всех объектов $r \in R$:

$$h(r) := \arg \min_h \sum_u (\hat{f}_{ur}(g(u), h) - f_{ur})^2;$$

George T., Merugu S. A scalable collaborative filtering framework based on co-clustering // 5-th IEEE int. conf. on Data Mining, 2005, Pp. 27–30.

Banerjee A., et al. A generalized maximum entropy approach to Bregman co-clustering and matrix approximation // 10-th KDDM, 2004, Pp. 509–514.

Матричные разложения

T — множество тем (интересов): $|T| \ll |U|$, $|T| \ll |R|$;
 p_{tu} — неизвестный профиль клиента u ; $P = (p_{tu})_{|T| \times |U|}$;
 q_{tr} — неизвестный профиль объекта r ; $Q = (q_{tr})_{|T| \times |R|}$;

Задача: найти разложение $f_{ur} = \sum_{t \in T} \pi_t p_{tu} q_{tr}$;

Матричная запись: $F = P^T \Delta Q$, $\Delta = \text{diag}(\pi_1, \dots, \pi_{|T|})$;

Вероятностный смысл: $\underbrace{p(u, r)}_{f_{ur} ?} = \sum_{t \in T} \underbrace{p(t)}_{\pi_t} \cdot \underbrace{p(u|t)}_{p_{tu}} \cdot \underbrace{q(r|t)}_{q_{tr}}$;

Методы решения:

SVD — сингулярное разложение (плохо интерпретируется);

NNMF — неотрицательное матричное разложение: $p_{tu} \geq 0$, $q_{tr} \geq 0$;

PLSA — вероятностный латентный семантический анализ.

Разреженный SVD (Singular Value Decomposition)

Обычный не разреженный SVD: $\|F - P^T Q\|^2 \rightarrow \min_{P, Q}$.

Разреженный SVD: $\sum_{(u,r) \in D} \underbrace{\left(f_{ur} - \sum_{t \in T} p_{tu} q_{tr} \right)^2}_{\varepsilon_{ur}} \rightarrow \min_{P, Q}$.

Метод стохастического градиента:

перебираем все $(u, r) \in D$ многократно в случайном порядке и делаем каждый раз градиентный шаг для задачи $\varepsilon_{ur}^2 \rightarrow \min_{p_u, q_r}$:

$$p_{tu} := p_{tu} + \eta \varepsilon_{ur} q_{tr}, \quad t \in T;$$

$$q_{tr} := q_{tr} + \eta \varepsilon_{ur} p_{tr}, \quad t \in T;$$

Tacáks G., Pilászy I., Németh B., Tikk D. Scalable collaborative filtering approaches for large recommendation systems // JMLR, 2009, No. 10, Pp. 623–656.

Разреженный SVD

Преимущества метода стохастического градиента:

- легко вводится регуляризация:

$$\varepsilon_{ur}^2 + \lambda \|p_u\|^2 + \mu \|q_r\|^2 \rightarrow \min_{p_u, q_r};$$

- легко вводятся ограничения неотрицательности:

$$p_{tu} \geq 0, \quad q_{tr} \geq 0 \quad (\text{метод проекции градиента});$$

- легко вводятся обобщение для ранговых данных:

$$\sum_{(u,r) \in D} \left(\beta(f_{ur}) - \sum_{t \in T} p_{tu} q_{tr} \right)^2 \rightarrow \min_{P, Q, \{\beta_f\}}.$$

- легко реализуются все виды инкрементности: добавление
 - ещё одного клиента u ,
 - ещё одного объекта r ,
 - ещё одного значения f_{ur} .
- высокая численная эффективность на больших данных;

Вероятностный латентный семантический анализ (PLSA)

Пусть T — множество тем (интересов);
 Вероятностная модель посещений [Hofmann, 1999]:

$$p(u, r) = \sum_{t \in T} p(t) p(u|t) q(r|t).$$

Задача максимизации правдоподобия по $p(t)$, $p(u|t)$, $q(r|t)$:

$$L(\Delta, P, Q) = \sum_{u,r} f_{ur} \ln p(u, r) \rightarrow \max.$$

при ограничениях нормировки:

$$\sum_{t \in T} p(t) = 1; \quad \sum_{u \in U} p(u|t) = 1; \quad \sum_{r \in R} q(r|t) = 1.$$

Тематические профили вычисляются по формуле Байеса:

$$p(t|u) = \frac{p(u|t) p(t)}{\sum_{s \in T} p(u|s) p(s)}; \quad q(t|r) = \frac{q(r|t) p(t)}{\sum_{s \in T} q(r|s) p(s)}.$$

Максимизация правдоподобия: EM-алгоритм

Сформировать начальные приближения $p(t)$, $p(u|t)$, $q(r|t)$;
Повторять итерации до сходимости:

- **E-шаг:** скрытые переменные H по формуле Байеса:

$$H(t|u, r) = \frac{p(t)p(u|t)q(r|t)}{p(u, r)};$$

- **M-шаг:** аналитическое решение задачи $L(\Delta, P, Q) \rightarrow \max$:

$$p(t) = \frac{S(t)}{S}; \quad S(t) = \sum_{u,r} f_{ur} H(t|u, r); \quad S = \sum_{u,r} f_{ur};$$

$$p(u|t) = \frac{1}{S(t)} \sum_r f_{ur} H(t|u, r);$$

$$q(r|t) = \frac{1}{S(t)} \sum_u f_{ur} H(t|u, r).$$

Вывод формул M-шага

Распишем Лагранжиан:

$$\mathcal{L} = \sum_{u,r} f_{ur} \ln p(u, r) - \nu \left(\sum_{t \in T} p(t) - 1 \right) - \sum_{t \in T} \lambda_t \left(\sum_{u \in U} p(u|t) - 1 \right) - \sum_{t \in T} \mu_t \left(\sum_{r \in R} q(r|t) - 1 \right).$$

$$\frac{\partial \mathcal{L}}{\partial p(t)} = \sum_{u,r} f_{ur} \frac{p(u|t)q(r|t)}{p(u, r)} - \nu = 0;$$

$$\sum_{u,r} f_{ur} \frac{p(u|t)q(r|t)p(t)}{p(u, r)} = \nu p(t) \Rightarrow \nu = \sum_{u,r} f_{ur} = S;$$

$$p(t) = \frac{1}{S} \sum_{u,r} f_{ur} H(t|u, r) = \frac{S(t)}{S};$$

Вывод формул M-шага (продолжение)

$$\frac{\partial \mathcal{L}}{\partial p(u|t)} = \sum_r f_{ur} \frac{p(t)q(r|t)}{p(u,r)} - \lambda_t = 0;$$

$$\sum_r f_{ur} \frac{p(t)q(r|t)p(u|t)}{p(u,r)} = \lambda_t p(u|t) \Rightarrow \lambda_t = \sum_{u,r} f_{ur} H(t|u,r);$$

$$p(u|t) = \sum_r f_{ur} H(t|u,r) / \sum_{u,r} f_{ur} H(t|u,r).$$

$$\frac{\partial \mathcal{L}}{\partial q(r|t)} = \sum_u f_{ur} \frac{p(t)p(u|t)}{p(u,r)} - \mu_t = 0;$$

$$\sum_u f_{ur} \frac{p(t)p(u|t)q(r|t)}{p(u,r)} = \mu_t q(r|t) \Rightarrow \mu_t = \sum_{u,r} f_{ur} H(t|u,r);$$

$$q(r|t) = \sum_u f_{ur} H(t|u,r) / \sum_{u,r} f_{ur} H(t|u,r).$$

Обобщения, модификации, применения

- Если $f_{ur} \in Z = \{1, 2, \dots, z_{\max}\}$ — рейтинги, то вместо $p(u, r) = P(f_{ur} \neq \emptyset)$ надо оценивать $(z_{\max} - 1)$ вероятностей $p_z(u, r) = P(f_{ur} \leq z)$, $z \in Z$;
- Иерархические профили: темы разбиваются на подтемы;
- Инкрементные алгоритмы: обработка потока данных D ;
- Учёт априорной информации через начальное приближение профилей:
 - тематический каталог объектов;
 - соц-дем (анкеты) клиентов;
- Унифицированный профиль объектов и клиентов;
- Долгосрочный и краткосрочный профили;
- Оценивание сходства по частям профиля.

Данные Яндекс (Интернет-математика 2005)

Исходные данные:

7 дней работы поисковой машины Яндекс; объём лога 3.6 Гб;
14 606 пользователей;
207 312 запросов;
1 972 636 документов было выдано;
129 600 документов были выбраны пользователями.

Фрагмент лога:

```
1098353321109615996 (номер пользователя)
  французская кухня (запрос) 1110473322 (время запроса) 113906 0
    http://www.nature1.ru/ (сайт или документ)
    http://www.kuking.net/c7.htm 1110473328 (время клика)
    http://www.cooking-book.ru/national/french/
  ...
  жаренное мясо в вине 1110473174 1349 0
  ...
...
```


Данные Яндекс (Интернет-математика 2005)

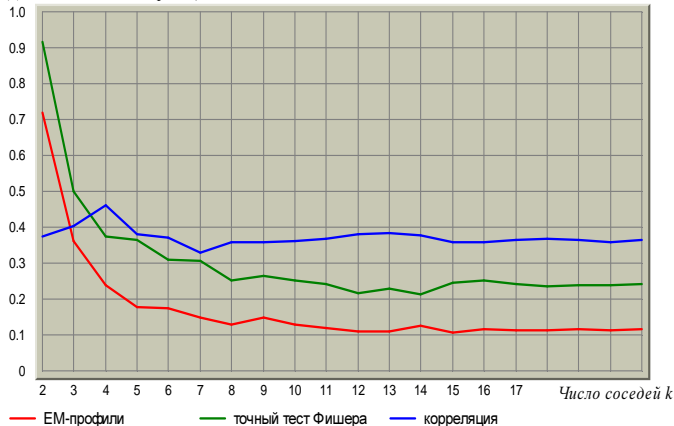
Схема эксперимента:

- Отбор наиболее посещаемых сайтов, $|R| = 1024$.
- Отбор наиболее активных пользователей, $|U| = 7300$.
- Введение критериев качества профилей:
 - 400 сайтов заранее классифицированы на $|T| = 12$ тематических классов;
 - Q_1 = доля неправильно восстановленных профилей;
 - Q_2 = число ошибок классификации методом kNN ;
- Оптимизация параметров по критерию качества.
- Построение профилей и оценок сходства сайтов.
- Визуализация: глобальные и локальные карты сходства.

Результаты: подбор меры сходства

оценки сходства по точному тесту Фишера (FET) лучше корреляций, а по профилям — ещё лучше!

Доля ошибок классификации методом kNN



Задача многомерного шкалирования (multidimensional scaling)

Дано: попарные расстояния R_{ij} между n объектами.

Найти: координаты этих объектов на плоскости $(x_i, y_i)_{i=1}^n$:

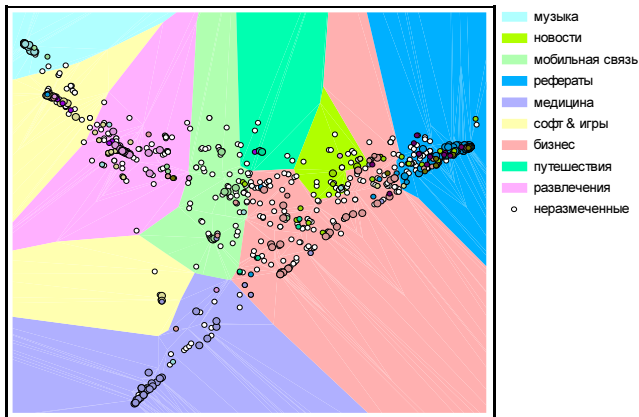
$$S = \sum_{i < j} \left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - R_{ij} \right)^2 \rightarrow \min_{(x_i, y_i)_{i=1}^n}$$

Карта сходства (Similarity Map) — это средство разведочного анализа многомерных данных:

- точечный график $(x_i, y_i)_{i=1}^n$;
- близким объектам соответствуют близкие точки;
- оси графика не имеют интерпретации;
- возможны искажения.

Карта поисковых интересов пользователей Рунета

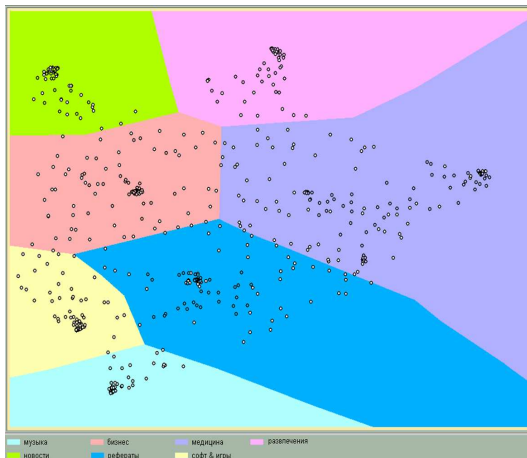
Многомерное шкалирование по FET-оценкам сходства, $|T| = 9$



Результат: темы удаётся проинтерпретировать :)

Карта поисковых интересов пользователей Рунета

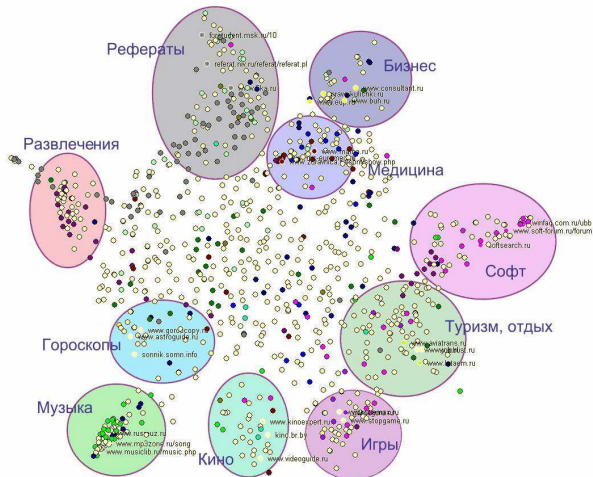
Многомерное шкалирование по профилям, $|T| = 7$



Результат кажется более содержательным :)

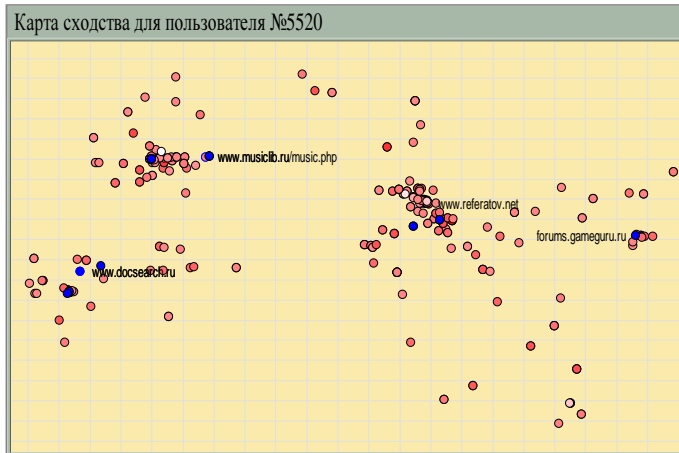
Карта поисковых интересов пользователей Рунета

Многомерное шкалирование по профилям, $|T| = 12$



Ещё одна визуализация: локальная карта пользователя

Визуальное представление персональных рекомендаций:



Резюме

Коллаборативная фильтрация (Collaborative Filtering) — это набор методов для построения рекомендательных систем (Recommender Systems).

Тематическое моделирование (Topic Modeling) — это набор методов для выявления латентных интересов клиентов или для выявления латентных тем в корпусе текстов.

Латентные модели обладают рядом преимуществ:

- тематические профили содержательно интерпретируемы, могут оцениваться по внешним данным,
- что позволяет решать проблему «холодного старта»
- и строить тематическую кластеризацию (таксономию);
- оценки сходства клиентов и объектов более адекватны;
- резко сокращается объём хранимых данных.