Применение коллаборативной фильтрации в задаче выделения селекторов.

Шинкевич Михаил

Московский физико-технический институт
Факультет управления и прикладной математики
Кафедра интеллектуальных систем

Научный руководитель д.ф.-м.н. К. В. Воронцов

Москва,
2015 г.

Описание сервиса

Имеется мобильный сервис А/Б-тестирования фотографий.

А/Б - тестирование

А/Б-тестирование (также говорят "сплит тестирование") заключается в сравнении двух версий (А и Б) одного объекта. Побеждает та версия, которая на одинаковых пользователях дает лучшую конверсию.

Проблема

С одной стороны, для получения достоверного результата тестирования необходимо, чтобы в тестировании приняло участие как можно большее **число пользователей**.
С другой стороны - для мобильного сервиса важна **скорость получения результата** тестирования.

Цель исследования

Разработать алгоритм персонализации рекомендаций, улучшающий качество сервиса.

Используемые методы

- Схожесть вкусов пользователей вычисляется при помощи **коллаборативной фильтрации**.
- Для кластеризации используется **FLAME* алгоритм**.

*FLAME(Fuzzy clustering by Local Approximation of Membership)

Литература

Кластеризация пользователей без использования социально-демографической информации:

- *Kwan Hui Lim, Amitava Datta* Detecting Communities with Common Interests on Twitter // School of Computer Science and Software Engineering. The University of Western Australia, 2013.
- *Zhang Y., Wu Y., Yang Q.* Community Discovery in Twitter Based on User Interests // Journal of Computational Information Systems, 2012. Vol. 8, № 3. P. 991–1000

Коллаборативная фильтрация:

- *Michael D. Ekstrand, John T. Riedl, Joseph A. Konstan* Collaborative Filtering and Recommender Systems // Computer Interaction, 2012. Vol. 4, № 2. P. 81–173

FLAME кластеризация:

- *Chattopadhyay S., Pratihar D. K., Sarkar S. C.* Fuzzy Clustering Algorithms // Computing and Informatics, 2011. Vol. 30. P. 701–720

Постановка задачи

Дано:

- Множество троек $T = \{u, c, r\}$ —
  {id пользователя, id А/Б-тестирования, выбор пользователя из А и Б}
- Функция схожести вкусов пользователей:

$$sim(u_i, u_j) = \alpha \frac{q_s^{ij}}{q_a^{ij}} + \beta \frac{w_s^{ij}}{q_a^{ij}} + \gamma \frac{s_b^{ij}}{s_a^{ij}},$$

q_s^{ij} - количество голосований, на которых оба пользователя i и j сделали одинаковый выбор,
q_a^{ij} - суммарное количество голосований, в которых принимали участие оба пользователя i и j,
w_s^{ij} - взвешенное количество голосований, на которых оба пользователя i и j сделали одинаковый выбор,
s_b^{ij} - количество голосований, на которых оба пользователя i и j отказались голосовать,
s_a^{ij} - суммарное количество голосований, на которых отказался голосовать хотя бы один из пользователей i и j.

Постановка задачи

Определение кластера

Множество пользователей разбито на непересекающееся множество кластеров при помощи FLAME алгоритма.
Предполагается, что пользователи одного кластера дают одинаковый ответ на одинаковых А/Б-тестированиях.

Найти:

Параметры α, β и γ, при которых доставляется минимум функции ошибки:

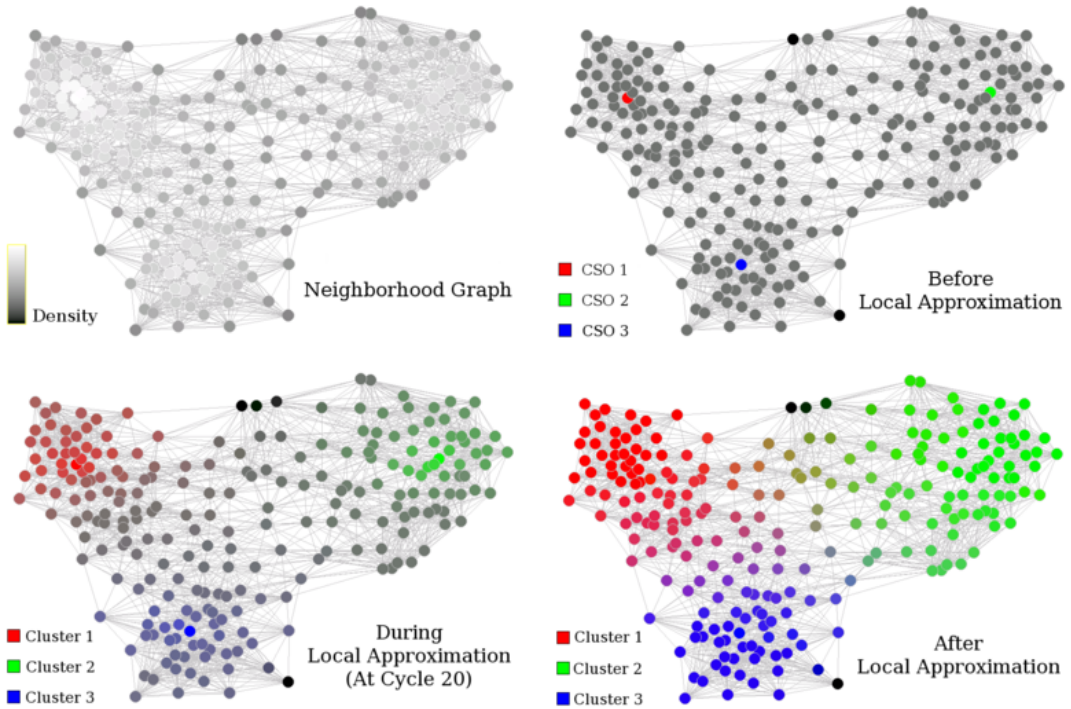
$$E(U) = \frac{1}{n}\sum_{i=1}^{n}[\frac{1}{m}\sum_{j=1}^{m}(\frac{out_{ij}}{all_i})],$$

out_{ij} - число пользователей из кластера i, сделавших в А/Б-тестировании j выбор не такой, как сделало большинство пользователей данного кластера,
all_i - число пользователей кластера i,
m - размер(количество А/Б-тестирований) тестовой выборки,
n - количество кластеров,
U - множество всех пользователей.

Описание FLAME алгоритма

Мотивация

FLAME - алгоритм мягкой кластеризации, базирующийся на k-ближайших соседях. Данный алгоритм был выбран в связи с тем, что он работает быстро (за суб-квадратичное время) и дает хорошие результаты для большой базы пользователей, что было показано в исследованиях, проведенных командой Twitter.



Оптимизационная задача FLAME

Необходимо минимизировать функцию:

$$E(\{\boldsymbol{p}\}) = \sum_{\boldsymbol{x} \in \boldsymbol{X}} \left\| \boldsymbol{p}(\boldsymbol{x}) - \sum_{\boldsymbol{y} \in \mathcal{N}(\boldsymbol{x})} w_{\boldsymbol{xy}} \boldsymbol{p}(\boldsymbol{y}) \right\|^2$$

$\boldsymbol{X}$ - множество объектов всех 3 типов,
$\boldsymbol{p}(\boldsymbol{x})$ - вектор мягкого членства объекта $\boldsymbol{x}$,
$\mathcal{N}(x)$ - множество ближайших соседей $\boldsymbol{x}$,
$w_{\boldsymbol{xy}}$ - коэффициенты, отражающие близость соседа, $\sum\limits_{\boldsymbol{y} \in \mathcal{N}(\boldsymbol{x})} w_{\boldsymbol{xy}} = 1$.

Функция ошибки может быть минимизирована решением следующих линейных уравнений:

$$p_k(\boldsymbol{x}) - \sum_{\boldsymbol{y} \in \mathcal{N}(\boldsymbol{x})} w_{\boldsymbol{xy}} p_k(\boldsymbol{y}) = 0, \quad \forall \boldsymbol{x} \in \boldsymbol{X}, \quad k = 1, ..., M$$

где M - число кластеров.
Следующая итеративная процедура может быть использована для решения этих линейных уравнений:

$$\boldsymbol{p}^{t+1}(\boldsymbol{x}) = \sum_{\boldsymbol{y} \in \mathcal{N}(\boldsymbol{x})} w_{\boldsymbol{xy}} \boldsymbol{p}^t(\boldsymbol{y})$$

Описание FLAME алгоритма

FLAME алгоритм состоит из 3 этапов:

1) Выделение структурной информации данных

1. Построить граф соседства соединяющий каждый объект с его k-ближайшими соседями.

2. Оценить плотность каждого объекта на основе близости к своим соседям

$$\rho_j = \frac{\max\limits_{u \in U}(\frac{1}{k}\sum_{i=1}^{k} d_{iu})}{\frac{1}{k}\sum_{i=1}^{k} d_{ij}}$$

3. Объекты относятся к одному из 3 типов:

a) Центр кластера: объекты с плотностью большей, чем все его соседи

b) Выбросы: объекты с плотностью меньшей, чем у всех соседей и ниже, чем заданный порог

c) Все остальные объекты

Описание FLAME алгоритма

2) Вычисление доли принадлежности каждому кластеру:

1. Инициализация членства

a) Каждому центру кластера присваивается полное членство своего кластера

b) Все выбросы присваиваются кластеру выбросов

c) Все остальные объекты получают равное членство каждого кластера

2. Итеративное обновление членства объектов на основе линейной комбинации членства своих соседей.

$$s_j = \sum_{i=1}^{k} (w_i s_i), \quad w_j = \frac{\frac{1}{d_j}}{\sum_{i=1}^{k} \frac{1}{d_i}}$$

3) Присвоение объектов к кластеру, в котором он имеет наибольшее членство

Описание решения

- На основе матрицы пользователь–А/Б-тестирование, в ячейках которой находится выбор(из А и Б) данного пользователя на данном тестировании, при фиксированных значениях параметров α, β и γ составляется матрица пользователь–пользователь, в ячейках которой хранится значение функции схожести данных пользователей.

- Применение FLAME кластеризации к матрице пользователь–пользователь

- Проверка качества кластеризации на тестовой выборке

Эксперимент

Для проведения эксперимента было взято $|U| = 1000$, $|C| = 10.000$,
U - множество пользователей,
C - множество А/Б-тестирований, в которых принимали участия все данные пользователи.

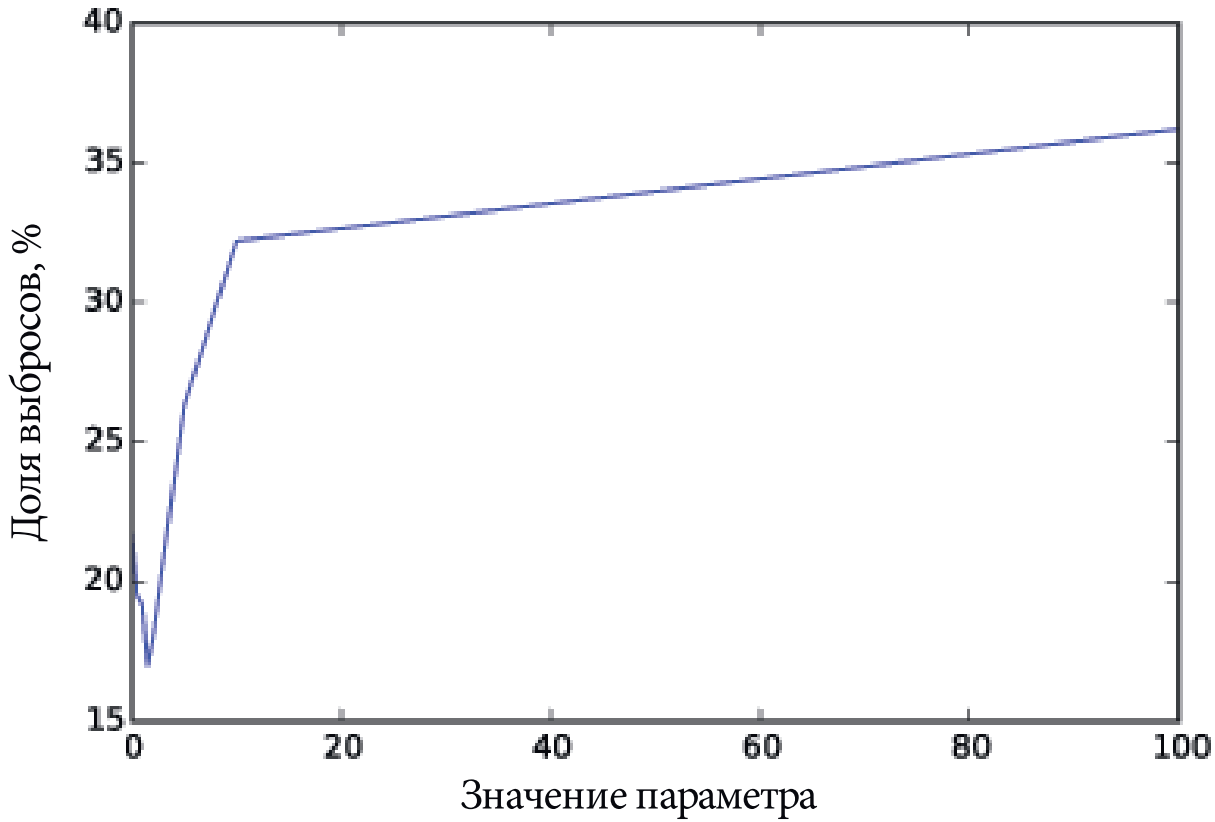
Данная выборка(множество опросов) была разбита на обучающую и тестовую в отношении 7 к 3.

Сначала коэффициенты значимости изучались по отдельности (т.е. остальные коэффициенты функции похожести пользователей занулялись).
Доли выбросов (значение функции $E(U) = \frac{1}{n}\sum_{i=1}^{n}[\frac{1}{m}\sum_{j=1}^{m}(\frac{out_{ij}}{all_i})]$):
$\alpha - 37\%$
$\beta - 26\%$
$\gamma - 24\%$

Далее коэффициенты изучались в комбинации (т.е. все коэффициенты уже брались ненулевыми). Фиксировались два коэффициента и варьировался третий.

Оптимизация α

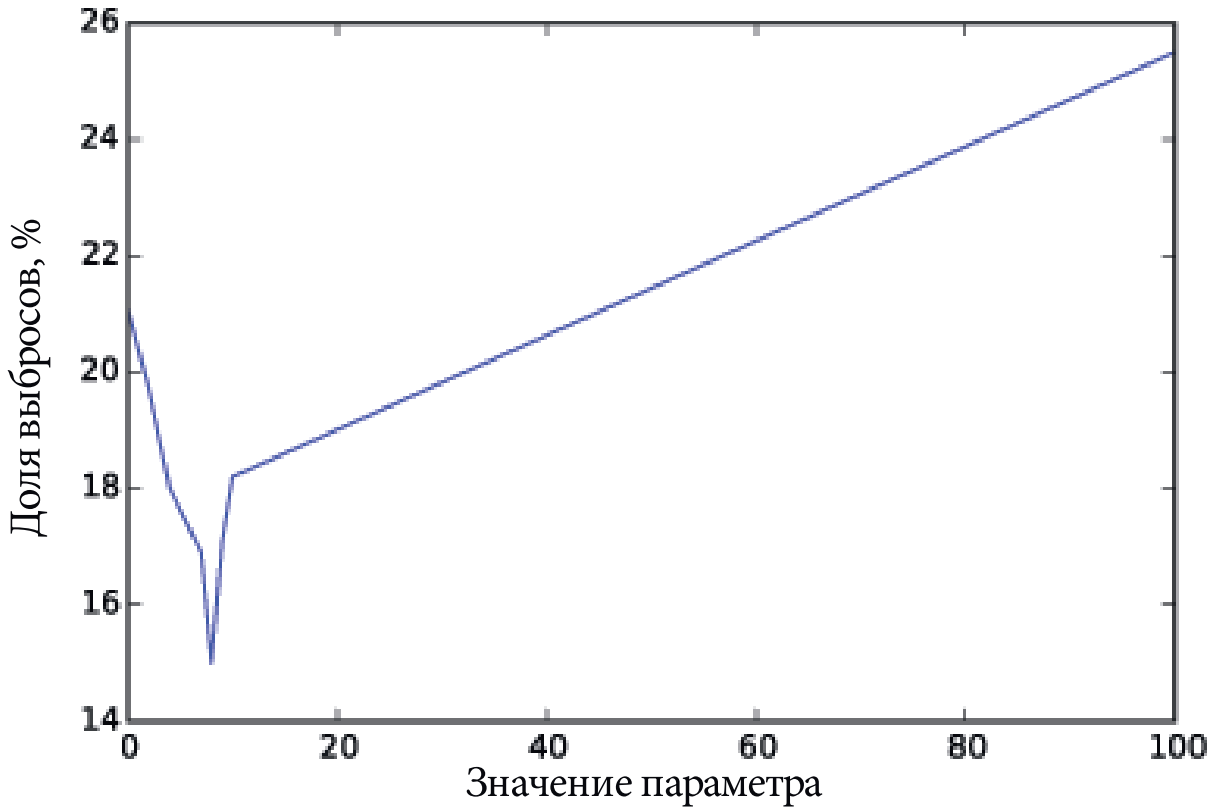
Сначала изучался коэффициент значимости α при фиксированных $\beta = 10$ и $\gamma = 10$.



(b) Оптимальный параметр α

Оптимизация β

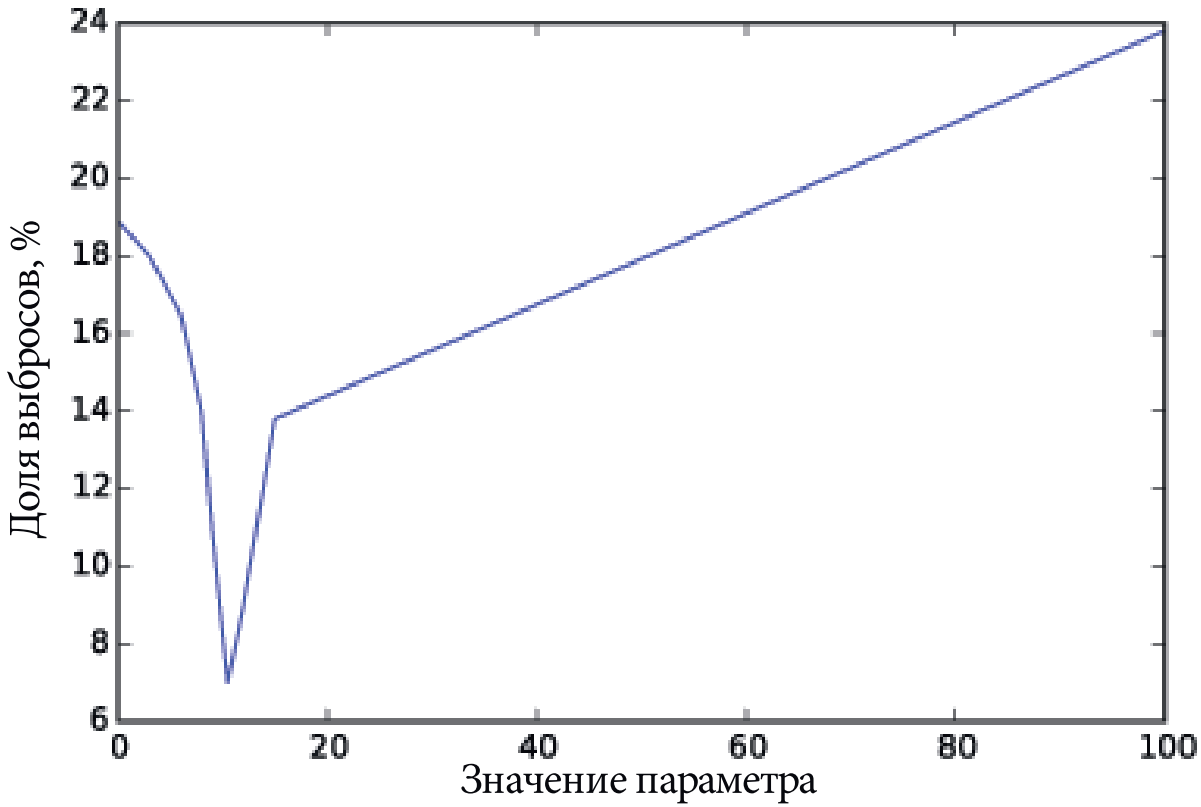
Далее изучался коэффициент значимости β при фиксированных $\alpha = 1,5$ и $\gamma = 10$.



(c) Оптимальный параметр β

Оптимизация γ

Далее изучался коэффициент значимости γ при фиксированных $\alpha = 1,5$ и $\beta = 8$.



(d) Оптимальный параметр γ

Результаты вычислительного эксперимента

Из полученных результатов видно, что:

- Ответ пользователей на опросы обусловлен их вкусами ($\alpha = 1,5$)
- Чем уникальнее ответ на опрос, тем больше это говорит о вкусе пользователя ($\beta = 8$)
- Отказ от голосования сильно свидетельствует о вкусе пользователя ($\gamma = 10,5$)

Заключение

В результате проделанной работы был получен алгоритм кластеризации пользователей по вкусам без использования их соц-демографической информации.

Это позволило увеличить скорость получения ответа пользователем в более, чем 8 раз, что помогло более, чем 400.000 людям.