



## Алгоритм машинного обучения для решения задачи формирования рекомендаций товаров и услуг

**Судаков В.А.\***

Московский авиационный институт (МАИ),  
г. Москва, Российская Федерация  
ИПМ им. М.В.Келдыша РАН, г. Москва, Российская Федерация  
ORCID: <https://orcid.org/0000-0002-1658-1941>  
e-mail: [sudakov@ws-dss.com](mailto:sudakov@ws-dss.com)

**Трофимов И.А.\*\***

Московский авиационный институт (МАИ),  
г. Москва, Российская Федерация  
e-mail: [trofimovc137@gmail.com](mailto:trofimovc137@gmail.com)

В статье предложен алгоритм машинного обучения без учителя для оценки наиболее возможных отношений между двумя элементами множеств клиентов и товаров/услуг с целью для построения рекомендательной системы. Рассмотрены методы на основе коллаборативной фильтрации и фильтрации на основе содержания. Разработан комбинированный алгоритм выявления отношений на множествах, сочетающий достоинства анализируемых подходов. Оценена сложность алгоритма. Даны рекомендации по эффективной реализации алгоритма с целью уменьшения объема используемой памяти. На примере задачи рекомендаций книг показано применение данного комбинированного алгоритма. Данный алгоритм может использоваться для «холодного старта» рекомендательной системы, когда ещё не существует размеченных качественных выборок обучения более сложных моделей.

**Ключевые слова:** машинное обучение, обучение без учителя, рекомендательные системы, сходство объектов, отношение, множество.

### Для цитаты:

Судаков В.А., Трофимов И.А. Алгоритм машинного обучения для решения задачи формирования рекомендаций товаров и услуг // Моделирование и анализ данных. 2020. Том 10. № 4. С. 5–16. DOI: <https://doi.org/10.17759/mda.2020100401>

\*Судаков Владимир Анатольевич, д.т.н., профессор каф.805, Московский авиационный институт (МАИ), ведущий научный сотрудник, Федеральное государственное учреждение «Федеральный исследовательский центр Институт прикладной математики им. М.В. Келдыша Российской академии наук» (ИПМ им. М.В.Келдыша РАН), ORCID: <https://orcid.org/0000-0002-1658-1941>, e-mail: [sudakov@ws-dss.com](mailto:sudakov@ws-dss.com)

\*\*Трофимов Иван Александрович, студент, Московский авиационный институт (МАИ), e-mail: [trofimovc137@gmail.com](mailto:trofimovc137@gmail.com)



## 1. ВВЕДЕНИЕ

В связи со стремительным распространением алгоритмов и методов машинного обучения в решении прикладных задач, на сегодняшний день появилось множество сфер их применения в бизнес-задачах повышения эффективности, получения прибыли, постановки прогнозов и так далее. Одним из направлений машинного обучения, которое сейчас особенно востребовано в прикладной области является сфера рекомендательных систем.

Увеличение роли рекомендательных моделей связано как с широким распространением интернет-продаж, а соответственно и интернет-рекламы, которая довольно адаптивна и может легко подстраиваться под конкретного пользователя, так и с наличием огромного количества информации, из которой можно тем или иным способом получить предпочтения клиента и сделать рекомендацию ориентированной под его специфические интересы и запросы [2].

Проблемой зачастую является качество самих исходных данных для построения прогноза. Эффективные модели требуют размеченных данных для обучения с учителем (например рейтинги, оставленные пользователями для каждого товара), либо наоборот, модели ориентируются лишь на схожесть товаров относительно друг друга, не учитывая опыт других пользователей.

В данной статье предлагается алгоритм построения ранжированных релевантных рекомендаций, основывающийся как на схожести пользователей, так и на сходстве товаров и услуг и оценивающий наиболее вероятные, но на данный момент ещё не появившиеся отношения между элементами двух исходных множеств.

## 2. ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ

На данный момент в рекомендательных системах существует два основных подхода создания рекомендаций – коллаборативная фильтрация и фильтрация на основе содержания [2].

Метод коллаборативной фильтрации использует данные об известных предпочтениях пользователей для прогнозирования предпочтений для других пользователей. Алгоритм основывается на допущениях о том, что пользователи, совместно оценившие схожим образом некоторую группу товаров, будут схожим образом оценивать и те товары, которые ещё не были оценены одним из пользователей [1]. Для наглядности данное предположение проиллюстрировано в таблице 1.

Таблица 1

Таблица коллаборативной фильтрации

	Товар № 1	Товар № 2	...	Товар № N
Пользователь № 1	5	4		4
Пользователь № 2	5			3
...				
Пользователь № M	?	4		4



Далее для активного прогнозируемого значения ячейки «пользователь – товар» формируется прогноз на основе информации о соседстве (фильтрация, основанная на соседстве) либо с помощью модели машинного обучения (фильтрация, основанная на модели). Прогнозируется значение – это значение, которое пользователь поставил-бы данному товару, если бы ему пришлось его оценивать.

Преимущества метода коллаборативной фильтрации:

- Метод является универсальным для данных подходящего формата.
- Для работы не требуется детального описания товара.
- В большинстве практических задач метод имеет высокую точность.

Недостатки метода коллаборативной фильтрации:

- Обучающие данные должны быть размечены (необходимо иметь оценки пользователей). Зачастую, при начале работы рекомендательной системы, нет данных о предпочтениях пользователей. Также имеющихся данных может быть недостаточно для оценки поведения пользователя и предсказания этого поведения (один из вариантов работы алгоритма) [2].
- Разреженность данных и масштабируемость. Многие пользователи могут не оставлять пользовательскую оценку (рейтинг) товара. Но даже если такие данные есть, то товаров, как правило огромное количество, как и пользователей, что приводит к тому, что обучающие данные крайне разрежены и обладают большой размерностью. Из этого вытекает как существенные затраты на вычисления (сложность алгоритма  $O(M*N)$ ), так и проблемы, связанные с понятием «проклятие размерности», при котором новые данные могут лежать крайне далеко от обучающих образцов.
- Проблема холодного старта. Трудно рекомендовать товары, недавно добавленные в систему, или формировать рекомендации для пользователей, которые ещё ничего не выбрали [3].
- Синонимия. Схожие объекты с различными именами могут быть рассмотрены системой как различные. Например «книги для детей» и «детская литература».
- Существуют пользователи, чье мнение не совпадает с мнением большинства и для них трудно что-либо рекомендовать [2].

Другой метод – это метод фильтрации на основе содержания. Он основан на наличии описания о рекомендуемых объектах. Описание состоит из некоторого набора характеристик элемента. Значения характеристик должны в том или ином виде существовать для начала работы алгоритма [4].

Каждому пользователю ставится в соответствие профиль, содержащий характеристики, схожие с характеристиками рекомендуемых объектов. Как правило пользователя прямо спрашивают о его предпочтениях (например, просят указать наиболее интересные жанры), либо профиль формируется из истории уже заказанных пользователем товаров.

Преимущества метода фильтрации на основе содержания:

- Не требуется наличие большой группы пользователей для формирования рекомендации.
- Новые элементы могут быть рекомендованы сразу, после добавления их характеристик.



Недостатки метода фильтрации на основе содержания:

- Необходимо подробное описание товаров (большое количество характеристик), для возможности наиболее точно их различать.
- Метод зависит от предметной области и полезность полученных рекомендаций ограничена.
- Характеристики профиля пользователя должны быть сопоставимы с характеристиками предметов.
- Метод выдает рекомендации из «ближайших соседей» предмета, не рекомендуя что-то новое.
- Метод не учитывает опыт других пользователей.

Как видно, основные подходы в построении рекомендаций обладают рядом жестких требований к данным [4]. Необходимо либо наличие рейтингов пользователей о товаре, либо подробное описание товара в виде характеристик и наличие профиля пользователя.

Представленный в данной статье метод использует идеи из обоих предыдущих методов, при этом предоставляя гораздо более низкие требования к качеству данных для обучения. При этом отсутствия каких-либо данных приведет к тому, что метод просто вырождается в частный случай подхода, основанный на близости пользователей (если нет никакой информации о товаре), либо в частный случай фильтрации на основе содержания (если невозможно определить какую-либо информацию о сходстве пользователей) [5]. Но как будет показано в части с практическим применением метода, достаточно исторических данных о том, какие товары заказывал пользователь.

Также преимуществом метода является то, что полученным рекомендациям будет поставлен в соответствие численный рейтинг рекомендации, что позволит упорядочить рекомендации по релевантности. При этом для этого не требуются оценки товаров пользователями.

### 3. ПОСТАНОВКА ЗАДАЧИ И АЛГОРИТМ РЕШЕНИЯ

Пусть имеется множество пользователей  $U = \{u_1, u_2, \dots, u_n\}$ , множество объектов  $O = \{o_1, o_2, \dots, o_m\}$  и множество отношений между пользователями и объектами  $C = \{c_1, c_2, \dots, c_x\}$ , где  $c_i = (u_j, o_k)$  – наличие отношения между пользователем  $u_j$  и объектом  $o_k$ . Требуется для пользователя  $u_x$  оценить ещё не существующие, но наиболее вероятные отношения с элементами множества  $O$ .

Описание алгоритма:

1. Оценка расстояний между всеми элементами множества  $U$  с помощью некоторой функции  $Fu(u_1, u_2)$ . Функция  $Fu$  будет оценивать «схожесть пользователей» относительно друг друга. Данная функция является одним из гиперпараметров модели. При наличии численных признаков  $Ux = \{ux_1, ux_2, \dots, ux_n\}$  – это может быть метрика в многомерном пространстве, либо Гаусовская радиальная функция близости. При отсутствии каких-либо признаков пользователей, в качестве меры близости используются функции пересечения множеств используемых



товаров, например расстояние Жаккарда. Далее, в практическом примере, используется метрика на основании пересечения множеств.

2. Оценка расстояний между всеми объектам рекомендаций  $O$ . Аналогично расстоянию между пользователями, вводится некоторая функция близости объектов  $Fo(o_1, o_2)$ . Данная функция является вторым гиперпараметром модели и выбирается в зависимости от поставленной задачи и пожеланий в результатах. В практическом примере в качестве такой функции использовалась косинусная схожесть векторизованных текстовых описаний объектов [6].
3. Оценивается численный показатель возможного отношения каждого объекта множества  $O$  с пользователем  $u_s$ , для которого строится прогноз, как взвешенная сумма существующих отношений, умноженных на коэффициенты близости пользователей и объектов, сопоставленных с этим отношением:

$$r_{oi}(u_s, o_p) = \sum_{c_i \in C} Fu(u_s, u_{c_i}) \cdot Fo(o_{c_i}, o_p), \text{ где:}$$

$c_i = (u_{c_i}, o_{c_i})$  – связь между пользователем  $u_{c_i}$  и объектом  $o_{c_i}$ ,

$u_s$  - пользователь, для которого составляется рекомендация,

$Fu(u_s, u_{c_i})$  – близость пользователя из  $c_i$  с пользователем  $u_s$ , для которого вычисляется прогноз,

$o_p$  - объект, для которого оценивается возможность существования связи с  $u_s$ ,

$Fo(o_{c_i}, o_p)$  – близость анализируемого объекта  $o_p$  и объекта  $o_{c_i}$ ,

$u_s$  - пользователь, для которого составляется рекомендация.

Вычисление возможности связи выполняется для всех пар  $(u_s, o_p)$ , где наличие этой связи ещё не подкреплено экспериментально:  $(u_s, o_p) \notin C$ .

4. Для каждого пользователя  $u_s$  объекты  $o_p$  ранжируются по полученному для них рейтингу  $r_{oi}(u_s, o_p)$ .

Данный алгоритм исходит из двух предположений об имеющихся данных:

- Пользователи образуют группы, в которых интересы близких пользователей совпадают.
- Объекты образуют группы, близкие объекты в которых объединены общей тематикой и интересуют схожих пользователей.

Для релевантных результатов алгоритма достаточно выполнения хотя-бы одного из этих предположений.

При отсутствии схожих групп пользователей, их связи будут учитываться с малым коэффициентом (из-за удаленности друг от друга). Либо не будут иметь пересечений между собой. В таком случае модель вырождается в рекомендации товаров, наиболее похожих на те, которые пользователь уже покупал.

При отсутствии схожести в группах товаров аналогичным образом товары, не имеющие связей с группой ближайших пользователей, будут иметь малые веса либо не будут пересекаться. В таком случае модель вырождается в рекомендации тех товаров, которые наиболее популярны у схожих пользователей (по количеству связей, умноженных на близость пользователей).

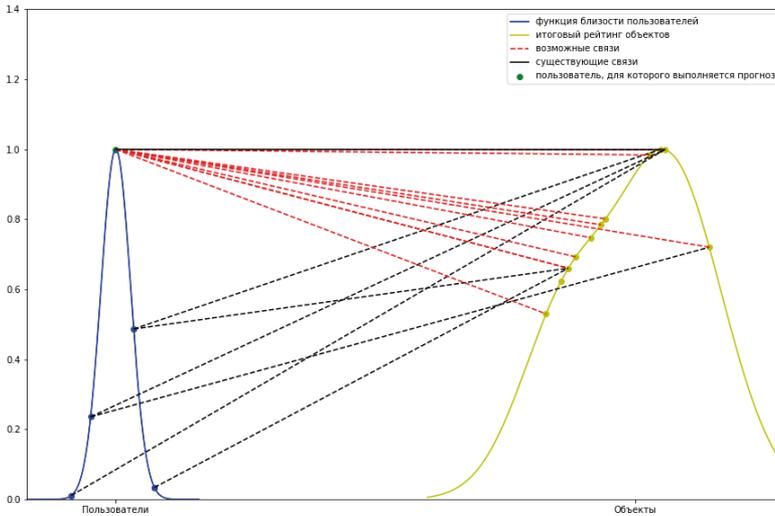


Рис. 1. Пример отношения между множествами

На рисунке 1. Приведен пример построения прогнозируемых отношений между множеством пользователей и множеством объектов. В данном примере у пользователей и объектов есть ряд численных признаков, а близость между объектами определяется Гаусовой радиальной функцией. Синим графиком отмечена «близость» прочих пользователей к тому, для которого строится рекомендация. Черными линиями отображены существующие связи между множествами. Желтый график – итоговый рейтинг каждого объекта для рекомендации пользователю. Красный пунктир – возможные рекомендации. Каждая рекомендация имеет свой рейтинг, по которому может проведено ранжирование.

#### 4. СЛОЖНОСТЬ АЛГОРИТМА

Первый шаг алгоритма имеет сложность  $O\left(\frac{M^2}{2} - M\right)$  для расчета схожести пользователей если  $F_{ij}$  – симметричная функция, в противном случае  $O(M^2 - M)$ . А второй шаг алгоритма имеет сложность  $O\left(\frac{N^2}{2} - N\right)$  для расчета схожести товара.

При полном расчете приходится хранить матрицы соответствующего размера, что может потребовать больших затрат памяти. Для решения проблемы с памятью можно поступить следующим образом: для пользователей и объектов рассчитать схожести внутри случайно-взятых подмножествах выборки. Определить квантили  $q_u$  и  $q_o$  для метрики близости внутри объектов и учитывать только тех соседей, у которых степень «близости» выше данного порогового значения. Таким образом можно хранить лишь таблицу вида: «объект-множества»: «ближайшие соседи и их близость».



Это значительно сэкономит используемую память (даже по сравнению с разреженными массивами, т.к. большинство объектов имеют очень низкую, но не нулевую близость), а для формирования прогноза надо будет сделать лишь следующие действия:

- взять ближайших соседей пользователя,
- взять соответствующие их связям объекты,
- из таблицы объектов выбрать ближайших соседей к объектам,
- рассчитать ранги всех объектов как взвешенные суммы произведений функций близости.

В данном случае квантили представляют собой ещё два гиперпараметра модели, которые можно подобрать в зависимости от задачи.

## 5. ПРИМЕР ПРАКТИЧЕСКОГО ИСПОЛЬЗОВАНИЯ АЛГОРИТМА

Данный алгоритм был использован при решении задачи формирования рекомендации читателям библиотек по имеющимся данным о выдаче книг. Эта задача была представлена на хакатоне «Лидеры цифровой трансформации» от департамента культуры города Москвы. Задача включала в себя несколько разделов, в качестве примера приведены рекомендации по книгам.

Исходные данные содержали:

- историю пользователей в формате «id пользователя» и «id взятых книг»,
- каталог книг, содержащий краткую текстовую информацию о книге или журнале: название, автор, издательство, жанр, возрастные ограничения.

В данных отсутствуют профили пользователей, рейтинги или предпочтения пользователей, а также нет критериев того, что считается оптимальной рекомендацией. Также требованием является наличие в рекомендациях объектов, не похожих на то, что он читал ранее.

Данные особенности накладывают существенные ограничения на использование базовых алгоритмов рекомендательных систем. Отсутствуют рейтинги пользователей. Интерпретация взятия книги как рейтинга  $\langle 1 \rangle / \langle 0 \rangle$  приводит к не релевантному результату рекомендации. Информации о объектах хватает лишь для разбиения на крупные множества.

Данные особенности делают задачу хорошим вариантом для тестирования предложенного метода.

Определение гиперпараметров модели:

1. В качестве функции близости пользователей была определена функция отношения пересечения множеств взятых книг ко множеству книг, взятых первым пользователем: Какую долю из книг первого пользователя уже брал второй. Данная функция не является симметричной. Функция используется для более сильной ориентации в прогнозе на пользователей, которые брали большее количество книг. В тоже время пользователь, имеющий небольшой «опыт» использования системы будет иметь меньшую схожесть.



2. В качестве функции близости между объектами (книгами) использовалась косинусная похожесть векторизированных описаний книг.
3. В модели не производился полный расчет таблиц, применялась оптимизация по памяти, описанная ранее. Квантиль для пользователей  $q_u = 0.7$ , квантиль для схожести книг  $q_o = 0.8$ .
4. После определения границы ближайших соседей, функции близости были упрощены до ступенчатых (1 если объект входит в границу).

Параметры были оценены при обучении на подвыборке из  $N = 2000$  пользователей и  $M = 5000$  книг.

Примеры рассмотрим примеры рекомендаций, выдаваемых моделью для разных пользователей.

Прочитанные первым случайным пользователем книги: автор – «Устинова Татьяна Витальевна», название – «Пять шагов по облакам», жанр – «Художественная литература»; автор – «Устинова Татьяна Витальевна», название – «Седьмое небо», жанр – «Художественная литература». Топ 5 рекомендаций, рассчитанных по модели пользователя, вошли книги, показанные на рисунке 2.

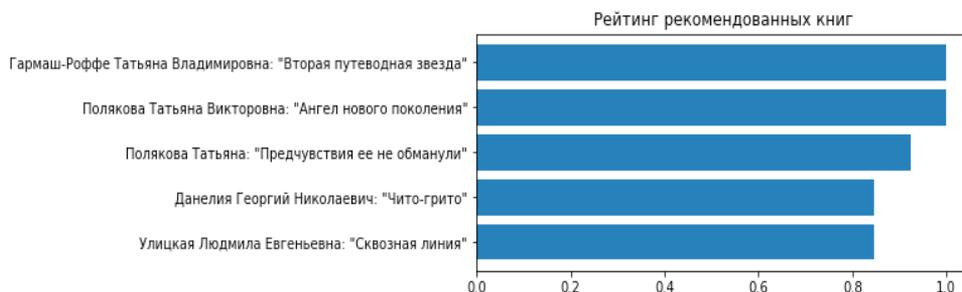


Рис. 2. Пример рекомендаций для первого случайного пользователя

Для пользователя, взявшего всего одну книгу: автор – «Тимм Уве», название – «Руди-Пятачок», жанр – «Художественная литература», топ 10 рекомендаций, рассчитанных по модели пользователя, вошли книги, показанные на рисунке 3.

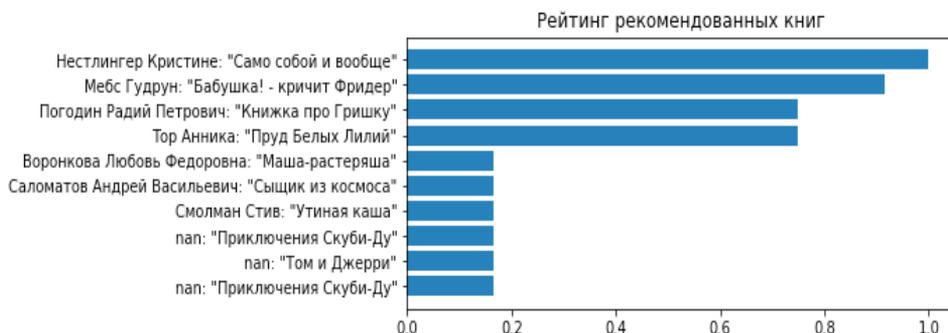


Рис. 3. Пример рекомендаций для второго случайного пользователя

Для пользователя с узкоспециализированным запросом были взяты книги: автор – «Пушкарева Наталья Львовна», название – «Частная жизнь русской женщины XVIII века», жанр – «Этнология (этнография, народоведение); автор – «Данелия Георгий Никола-евич», название – «Чито-грито», жанр: «Художественно-игровое кино». Топ 10 рекомендаций системы показан на рисунке 4.

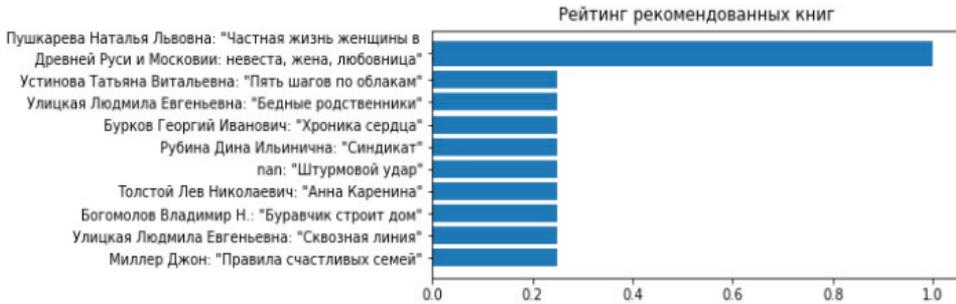


Рис. 4. Пример рекомендаций для пользователя с узкоспециализированным запросом

## 6. ЗАКЛЮЧЕНИЕ

Даже при существенном упрощении модели: ступенчатые функции близости, обучение лишь на малой части выборки, уже можно сказать, что модель показывает релевантные результаты рекомендаций. При необходимости данную модель можно настроить более тонким образом, используя более сложные функции близости и подбрав квантили, отвечающие за регуляризацию.

Преимущества модели:

- Низкие требования к качеству данных. Достаточно того, чтобы на одном из множеств было возможно определить функцию близости элементов множества.
- Обучение без учителя. Для модели не требуется размечать данные. Хватает лишь наличия отношений между множествами.
- Ранжированный результат. Модель выдает не только рекомендации, но и их ранг. При использовании сложных функций близости (таких например, как Гаусова радиальная функция RBF), полученные значения можно будет интерпретировать как уверенность в «удачной» рекомендации.
- Решена проблема холодного старта. Новый пользователь либо имеет связь с хотя-бы одним предметом, либо на множестве пользователей определена функция близости, не зависящая от «опыта» пользователя.
- За счет учета как опыта других пользователей, так и схожести объектов, генерируются разнообразные рекомендации.
- Частично решена проблема для «уникальных» пользователей, интересующихся узким кругом объектов. Рекомендации для них будут релевантны, если такие пользователи, либо объекты, интересующие их, формируют небольшую локальную группу. В таких случаях фактически рекомендации локализованы в подмно-



жестве основного множества объектов. Но при этом в рекомендации будут попадать и популярные для всех объекты, правда с меньшим рейтингом.

- Возможность оптимизации модели с помощью гиперпараметров.

Недостатки модели:

- Высокая вычислительная сложность, связанная с расчетом двух таблиц схожести.
- Низкая точность на пользователях, с большим «разбросом» интересов. Таким образом рекомендуются наиболее популярные объекты.
- Вычислительная сложность оперативного добавления новых элементов (необходимость считать схожесть с остальной выборкой), а в особенности при использовании несимметричных функций похожести (расчет схожести всей выборки с новым элементом)

Из совокупности имеющихся недостатков и преимуществ данного метода, можно сделать вывод, что модели подобного рода могут использоваться как модели «холодного старта» рекомендательной системы, т.е. являться «базовыми моделями» [2]. Когда ещё не существует размеченных качественных выборок для обучения более сложных моделей. На основе подобной модели можно составить первичные рекомендации, проверить их, оценить и сформировать выборку данных для более сложных моделей. Таких, например, как колаборативная фильтрация, в которой можно использовать полученные «базовой моделью» ранги [1].

Данный подход может использоваться и в других классах задач, не связанных с системами рекомендаций. Например, при анализе пользователей социальных сетей и сопоставления подмножеств групп пользователей друг с другом.

### *Литература*

1. *Melville P., Mooney R., Nagarajan R.* Content-Boosted Collaborative Filtering for Improved Recommendations // University of Texas, USA : Материалы конф. / AAAI-02, Austin, TX, USA, 2002. – 2002. – pp. 187–192.
2. *Jannach D., Zanker M., Felfering A., Friedrich G.* Recommender Systems: An Introduction. Cambridge University Press, 2010.
3. *Ricci F., Rokach L., Shapira B., Kantor P.* Recommender Systems: Handbook. Springer, 2011.
4. *Linden G., Smith B., York J.* Amazon.com recommendations: item-to-item collaborative filtering // Internet Computing – IEEE 7 2003 – pp. 76–80.
5. *Melville P., Mooney R.J., Nagarajan R.* Content-boosted collaborative filtering for improved recommendations // in Proceedings of the National Conference on Artificial Intelligence – 2002 – pp. 187–192.
6. *Белова К.М., Судаков В.А.* Исследование эффективности методов оценки релевантности текстов // Препринты ИПМ им. М.В. Келдыша 2020. No 68. 16 с. <http://doi.org/10.20948/prepr-2020-68>



# The Machine Learning Algorithm for Solving the Problem of Generating Recommendations for Goods and Services

**Vladimir A. Sudakov\***

Moscow Aviation Institute (MAI), Moscow, Russian Federation  
Keldysh Institute of Applied Mathematics (Russian Academy of Sciences),  
Moscow, Russian Federation  
ORCID: <https://orcid.org/0000-0002-1658-1941>  
e-mail: sudakov@ws-dss.com

**Ivan A. Trofimov\*\***

Moscow Aviation Institute (MAI), Moscow, Russian Federation  
e-mail: trofimovc137@gmail.com

The article proposes an unsupervised machine learning algorithm for assessing the most possible relationship between two elements of a set of customers and goods / services in order to build a recommendation system. Methods based on collaborative filtering and content-based filtering are considered. A combined algorithm for identifying relationships on sets has been developed, which combines the advantages of the analyzed approaches. The complexity of the algorithm is estimated. Recommendations are given on the efficient implementation of the algorithm in order to reduce the amount of memory used. Using the book recommendation problem as an example, the application of this combined algorithm is shown. This algorithm can be used for a “cold start” of a recommender system, when there are no labeled quality samples of training more complex models.

**Keywords:** machine learning, unsupervised learning, recommender systems, object similarity, relation, set.

## **For citation:**

Sudakov V.A., Trofimov I.A. The Machine Learning Algorithm for Solving the Problem of Generating Recommendations for Goods and Services. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 4, pp. 5–16. DOI: <https://doi.org/10.17759/mda.2020100401> (In Russ., abstr. in Engl.).

\* **Vladimir A. Sudakov**, Doctor of Technical Sciences, Professor of Department 805, Moscow Aviation Institute (MAI), Leader Researcher, Keldysh Institute of Applied Mathematics (Russian Academy of Sciences), ORCID: <https://orcid.org/0000-0002-1658-1941>, e-mail: sudakov@ws-dss.com

\*\* **Ivan A. Trofimov**, student, Moscow Aviation Institute (MAI), e-mail: trofimovc137@gmail.com



### **References**

1. Melville P., Mooney R., Nagarajan R. Content-Boosted Collaborative Filtering for Improved Recommendations. University of Texas, USA. *Proceeding of AAAI-02*, Austin, TX, USA, 2002. – 2002. – pp. 187–192.
2. Jannach D., Zanker M., Felfering A., Friedrich G., *Recommender Systems: An Introduction*. Cambridge University Press, 2010.
3. Ricci F., Rokach L., Shapira B., Kantor P. *Recommender Systems: Handbook*. Springer, 2011.
4. Linden G., Smith B., York J., Amazon.com recommendations: item-to-item collaborative filtering. *Internet Computing – IEEE 7 2003* – pp. 76–80.
5. Melville P., Mooney R.J., Nagarajan R. Content-boosted collaborative filtering for improved recommendations. *Proceedings of the National Conference on Artificial Intelligence – 2002* – pp. 187–192.
6. Belova K.M., Sudakov V.A. Issledovanie effektivnosti metodov ocenki relevantnosti tekstov [Research of the effectiveness of methods for assessing the relevance of texts]. *Preprinty IPM im. M.V. Keldysha = Keldysh Institute preprints*, 2020. No 68. 16 p. <http://doi.org/10.20948/prepr-2020-68>. (In Russ.).