Сложность восстановления матриц рейтингов рекомендательных систем

А. О. Савинский

В работе строится математическая модель рекомендательных систем. Приводятся оценки числа типов пользователей в матрице пользователей. Доказывается NP-трудность задач поиска представителя и перечисления матриц пользователей в общем случае и приводится полиномиальный алгоритм поиска представителя матриц пользователей в случае, когда число типов пользователей равно двум.

Ключевые слова: рекомендательные системы, восстановление матриц, пополнение матриц, поиск представителя матриц пользователей, перечисление матриц пользователей, оценка числа типов пользователей, NP-трудность.

Введение

Задача восстановления объекта или его свойств по частичной информации — одна из центральных в математике. В теории автоматов раздел, в котором эта задача ставится и решается, называется экспериментами с автоматами. Это обширное и активно развивающееся направление широко отражено в литературе [1, 2, 3, 4, 5, 6] и посвящено тому, что по некоторым вход-выходным последовательностям автомата делается попытка определить к какому из наперед заданных классов относится автомат. Задача восстановления объекта или его свойств по частичной информации возникаем при прогнозировании сверхсобытий, когда по частичной информации сверхслова из заданного сверхсобытия нужно определить это сверхслово или определить некоторые его свойства [7]. В теории дискретных функций данная проблематика рассматривается в разделе, называемом расшифровкой функций, который также относят к машинному обучению [8, 9, 10, 11, 12, 13, 14]. При расшифровке функций стараются за как можно меньшее число запросов на значение функции полностью восстановить всю функцию из некоторого наперед заданного класса. В такой постановке расшифровка называется точной (exact learning). Исследуется также вероятно примерно точноя расшифровка (probably approximately correct learning, PAC) [15, 16, 17], в которой наборы с известными значениями функции выбираются случайно. В этой постановке задача восстановления функций более близка к задаче восстановления матриц, возникающих в рекомендательных системах.

Рекомендательные системы — это системы, которые пытаются предсказать, какие объекты (музыкальные произведения, фильмы, книги, товары, новости, веб-сайты) понравятся пользователю[18, 19, 20]. Обычно в рекомендательных системах имеется большое количество пользователей и объектов, и для каждого пользователя система должна предложить свой персонализированный список рекомендуемых объектов. Связь между пользователями и объектами осуществляется с помощью матрицы рейтингов. Строки матрицы можно воспринимать как пользователей, а столбцы — как объекты. На пересечении i-ой строки и j-го столбца матрицы стоит либо оценка, которую поставил i-ый пользователь j-му объекту, либо специальный символ, который означает, что i-ый пользователь еще не оценил j-ый объект. Существует два основных подхода в создании рекомендательных систем: фильтрация содержимого[21, 22, 23] и коллаборативная фильтрация[24, 25, 26, 27, 28]. Но, как правило, на практике используется гибридный подход[29].

Фильтрация содержимого основана на анализе информации об объектах и пользователях. Задачи, которые возникают в данном подходе, связаны с анализом содержимого объектов [30, 31, 32]. В данном подходе имеет большое значение с каким типом объектов мы работаем. У рекомендательных систем, основанных на таком подходе, не возникаем проблем, когда матрица рейтингов сильно разрежена или когда в системе появляется новый пользователь или объект, в отличии от рекомендательных систем, основанных на коллаборативной фильтрации. Но такой подход требует больших затрат для определения всех необходимых характеристик пользователей и объектов. Данный подход используется в проекте *Music Genome Project*: музыкальный аналитик оценивает каждую композицию по сотням различных музыкальных характеристик, которые могут использоваться для выявления музыкальных предпочтений пользователя[33].

Более интересным с математической точки зрения является подход, основанный на коллаборативной фильтрации. Данный подход основан на анализе матрицы рейтингов и предположениях, которые состоят в следующем: похожие пользователи ставят похожие оценки, и похожим объектам ставят похожие оценки. Этот метод хорошо работает, когда в системе имеется большое количество оценок. Такой подход используется в интернет-портале музыкальной тематики *last.fm*[34].

Одной из главных задач рекомендательных систем является задача восстановления матрицы рейтингов (matrix completion problem)[35, 36, 37, 38, 39, 40, 41]. Если не делать никаких дополнительных предположений о строении матрицы, то в общем случае восстановить неизвестные элементы матрицы невозможно. В зависимости от решаемой задачи, удобно вводить те или иные ограничения на восстанавливаемую матрицу. Одной из таких задач является задача восстановления матрицы при условии, что восстановленная матрица должна иметь наименьший ранг среди всех матриц, которые могут быть получены из исходной матрицы путем доопределения неизвестных элементов. Такая задача является NP-трудной, и все известные алгоритмы, которые дают точный результат, в теории и на практике имеют временную сложность равную двойной экспоненте от размерности матрицы[35]. Другая задача, которая возникает в управлении космическими аппаратами, является задача восстановления матрицы при условии, что полученная матрица будет устойчивой. Эта задача также является сложной[40, 41].

В данной работе строится математическая модель рекомендательных систем, основанных на коллаборативной фильтрации. Предполагается, что в рекомендательной системе существует всего несколько типов пользователей. Каждый тип пользователей однозначно определяется своим вектором оценок. Множество пользователей системы однозначно задается матрицей пользователей, строками которой являются векторы типов пользователей. Каждая строка матрицы соответствует пользователю системы. В данной модели матрица рейтингов — это матрица, получаемая из матрицы пользователей путем затирания некоторых элементов.

В данной работе получены оценки на число типов пользователей в матрице пользователей. Рассматриваются два типа задач: задача поиска представителя и задача перечисления матриц пользователей. В задаче поиска представителя матриц пользователей ищется любая матрица пользователей, которая порождает данную матрицу рейтингов. В задаче перечисления матриц пользователей ищутся все матрицы пользователей, которые порождают данную матрицу рейтингов. Когда число типов пользователей больше двух, доказывается NP-трудность этих задач. В случае с двумя типами пользователей приводится алгоритм, который за полиномиальное время решает задачу поиска представителя матриц пользователей.

Автор выражает глубокую благодарность научному руководителю д.ф.-м.н., профессору Э.Э. Гасанову за постановку задачи и внимание к работе. Также автор выражает благодарность профессору школы бизнеса Стерна Нью-Йоркского университета А.С. Тужилину за внимание к работе.

Математическая модель рекомендательных систем

Пусть фиксировано некоторое множество A и числа n и m. Множество A будем называть *алфавитом оценок*, n — *числом объектов*, m — *числом пользователей*. Множество C векторов длины n над алфавитом A будем называть *множеством типов пользователей*. Мощность множества C будем называть *числом типов пользователей*. Число типов пользователей множества C будем обозначать через M.

Матрицу U размера m на n, строки которой являются векторы из множества типов пользователей C, будем называть *матрицей пользователей* и говорить, что матрица U состоит из пользователей типа C или что множество C *порождает* матрицу U. Будем говорить, что число типов пользователей в матрице пользователей U не больше чем M.

В данной работе будем рассматриваться модель с затиранием, когда матрица рейтингов получается из матрицы пользователей путем затирания некоторых элементов. Рассмотрим множество B равное $A \cup \{\lambda\}$, где λ не принадлежит алфавиту A. Множество B будем называть *алфавитом рейтингов*. Элемент λ будем называть затертым элементом. Матрицу R размера m на n, полученную из матрицы пользователей U заменой некоторых элементов матрицы на символ λ, будем называть *матрицей рейтингов* и говорить, что матрица R *порождена* матрицей пользователей U.

Будем говорить, что два вектора одинаковой длины $x = (x_1, x_2, \dots, x_n)$ и $y = (y_1, y_2, \dots, y_n)$ над алфавитом B *смежны*, если для любого $i = 1, \dots\ , n$ выполнено одно из трех условий: $x_i = \lambda$, $y_i = \lambda$ или $x_i = y_i$. Смежность двух векторов x и y над алфавитом B означает, что существует вектор c над алфавитом B, что векторы x и y получаются из вектора c путем затирания некоторых элементов. При этом будем говорить, что векторы x и y *порождаются* вектором c.

Под *системой векторов* будем понимать занумерованные векторы, среди которых могут быть равные векторы.

Утверждение 1. *Система попарно смежных векторов над алфавитом B порождается одним вектором.*

Доказательство. Пусть S — система попарно смежных векторов длины n. Построим вектор c, который будет порождать систему S.

Рассмотрим i-ые координаты векторов из S, $i \in \{1, 2, \dots, n\}$. Среди значений координат не может встретиться два различных элемента из алфавита A, так как в противном случае соответствующие им векторы были бы не смежны. Если среди i-ых координат встретится элемент из алфавита A, то положим i-ую координату вектора c равной этому элементу. Иначе все координаты являются затертыми элементами, и i-ую координату вектора c положим равной λ.

Проделав данную операцию для всех $i = 1, \dots, n$, получим искомый вектор c над алфавитом B. □

Замечание 1. *Вектор c из утверждения 1 задает множество всех векторов над алфавитом A, которые порождают систему попарно смежных векторов, где в векторе c вместо элемента λ может стоять любой элемент из алфавита A.*

Вектор c над алфавитом B, порождающий систему попарно смежных векторов S, что любой другой вектор, порождающий систему S, получается из вектора c заменой некоторых элементов λ на элементы из алфавита A, будем называть *главным порождающим вектором* системы S. Вектор c построенный в доказательстве утверждения 1 является главным порождающим вектором S. Легко видеть, что для любой системы попарно смежных векторов существует единственный главный порождающий вектор.

Граф отличимости матрицы рейтингов

Сопоставим матрице рейтингов R граф G. Множество вершин графа G совпадает с множеством строк матрицы R. Две вершины графа G смежны тогда и только тогда, когда соответствующие им строки матрицы R не являются смежными. Полученный граф G будем называть *графом отличимости* матрицы R.

Правильной раскраской вершин графа в k цветов называется такое сопоставление каждой вершине графа одного из k цветов, что любая пара смежных вершин раскрашена в разные цвета.

Утверждение 2 (критерий раскрашиваемости вершин графа отличимости матрицы рейтингов)**.** *Пусть G — граф отличимости матрицы рейтингов R. Существует правильная раскраска вершин графа G в k цветов тогда и только тогда, когда существует матрица пользователей U, порождающая матрицу R, с числом типов пользователей не больше чем k.*

Доказательство. Пусть существует раскраска вершин графа G в k цветов. Рассмотрим любую такую раскраску. Тогда множество вершин графа G разбивается на непересекающиеся классы вершин, покрашенные в один цвет. Поскольку любые две вершины графа, покрашенные в один цвет не смежны, то соответствующие им строки матрицы R являются смежными. Следовательно, вершины графа G из одного класса соответствуют попарно смежным строкам матрицы R.

Из замечания 1 следует, что каждому классу смежных строк матрицы R можно сопоставить вектор над алфавитом A, который порождает строки из этого класса. Доопределим строки матрицы R из одного класса до одной и той же строки над алфавитом A. Получим матрицу пользователей U. Число различных строк в матрице U будет не больше k. Следовательно, число типов пользователей в матрице U будет не больше чем k.

Докажем в обратную сторону. Пусть существует матрица пользователей U, порождающая матрицу R, с числом типов пользователей не больше k, т.е. число различных строк матрицы U не превосходит k. Разобьем множество строк матрицы U на k непересекающихся классов, чтобы в каждом классе содержались равные строки. Множеству равных строк матрицы U соответствует множество попарно смежных строк матрицы R. А множеству попарно смежных строк матрицы R соответствует независимое множество вершин графа G, т.е. множество попарно несмежных вершин.

Следовательно каждому классу равных строк матрицы U соответствует независимое множество вершин графа G. Таким образом, мы разбили множество вершин графа G на k независимых множеств, и вершины из одного множества раскрасим в один цвет. Получим правильную раскраску вершин графа G в k цветов. $\square$

Оценка числа типов пользователей

Утверждение 3 (оценка сверху числа типов пользователей). *Пусть алфавит оценок $|A| < \infty$, n — число объектов. Тогда число типов пользователей M не больше чем $|A|^n$.*

Доказательство. Утверждение очевидным образом следует из того, что число различных векторов над алфавитом A длины n равно $|A|^n$. $\square$

Следствие 1. *Пусть n — число объектов, m — число пользователей. Если $|A| < \infty$, то число типов пользователей в матрице пользователей не больше чем $\min(m, |A|^n)$. Если $|A| = \infty$, то число типов пользователей в матрице пользователей не больше чем m.*

Хроматическим числом графа называется минимальное число красок, с помощью которых можно правильным образом раскрасить вершины графа.

Теорема 1 (оценка снизу числа типов пользователей). *Для любой матрицы пользователей U, порождающей матрицу рейтингов R, число типов пользователей в матрице U не меньше хроматического числа графа отличимости G матрицы R. Существует такая матрица пользователей U', порождающая матрицу R, что число типов пользователей в матрице U' равно хроматическому числу графа G.*

Доказательство. Предположим противное. Пусть существует матрица пользователей U, порождающая матрицу рейтингов R, что число типов пользователей в матрице U меньше хроматического числа графа отличимости G матрицы R. Тогда из утверждения 2 следует, что существует раскраска вершин графа G в число цветов меньшего хроматического числа графа G. Противоречие.

Пусть k — хроматическое число графа G. Рассмотрим произвольную раскраску вершин графа G в k цветов. Из утверждения 2 следует, что существует матрица пользователей U', порождающая матрицу R, с числом типов пользователей не больше k. Т.к. для любой матрицы пользователей U, порождающей матрицу рейтингов R, число типов пользователей в матрице U не меньше хроматического числа графа отличимости G матрицы R, то число типов пользователей матрицы U' равно k. □

Задачи поиска представителя матриц пользователей

Пусть R — матрица рейтингов. Задачу нахождения матрицы пользователей U , которая порождает матрицу R, будем называть *задачей поиска представителя матриц пользователей для матрицы рейтингов R.*

Задачу нахождения матрицы пользователей U с числом типов пользователей не больше чем M, которая порождает матрицу R, будем называть *задачей поиска представителя матриц пользователей при ограничении на число типов пользователей для пары* (R, M).

Задачу нахождения матрицы пользователей U с числом типов пользователей равным M, которая порождает матрицу R, будем называть *задачей поиска представителя матриц пользователей с заданным числом типов пользователей для пары* (R, M).

Задачу нахождения матрицы пользователей U с минимальным числом типов пользователей, которая порождает матрицу R, будем называть *задачей поиска представителя матриц пользователей с минимальным числом типов пользователей для матрицы рейтингов* R.

Класс квадратных матриц рейтингов с нулями на главной диагонали и без нулей вне главной диагонали обозначим через D.

Утверждение 4. *Пусть $\{0,1\} \subseteq A$, m — число пользователей, M — натуральное число, $M < m$, матрица рейтингов R принадлежит классу D. Если существует матрица пользователей с числом типов пользователей равным M, которая порождает матрицу R, то существует матрица пользователей с числом типов пользователей равным M+1, которая порождает матрицу R.*

Доказательство. Пусть U — матрица пользователей с числом типов пользователей равным M, которая порождает матрицу R. Т.к. $M < m$, то в матрице U найдутся две равные строки. Выберем одну из них. Пусть эта строка имеет номер i. Заменим в этой строке все нулевые элементы кроме i-ого на 1. Получим матрицу пользователей с числом типов пользователей равным $M+1$, которая порождает матрицу R. $\square$

Следствие 2. *Пусть $\{0,1\} \subseteq A$, m — число пользователей, M — натуральное число, матрица рейтингов R принадлежит классу D. Если существует матрица пользователей с числом типов пользователей равным M, которая порождает матрицу R, то для любого числа M', что $M \leqslant M' \leqslant m$, существует матрица пользователей с числом типов пользователей равным M', которая порождает матрицу R.*

Следствие 3. *Пусть $\{0,1\} \subseteq A$, m — число пользователей, M — натуральное число, $M \leqslant m$, матрица рейтингов R принадлежит классу D. Тогда существует представитель матриц пользователей при ограничении на число типов пользователей для пары (R, M), тогда и только тогда, когда существует представитель матриц пользователей с заданным числом типов пользователей для пары (R, M).*

Теорема 2. *Пусть $\{0,1\} \subseteq A$, матрица рейтингов R принадлежит классу D, M — натуральное число, $M > 2$. Задача определения, существует ли представитель матриц пользователей при ограничении на число типов пользователей для пары (R, M), является NP-полной.*

Доказательство. Очевидно, что задача определения, существует ли представитель матриц пользователей при ограничении на число типов пользователей, лежит в классе NP. Пусть нам дан граф G. Сведем NP-полную задачу о существовании раскраски вершин графа G в k, $k > 2$, цветов к задаче определения, существует ли представитель матриц пользователей при ограничении на число типов пользователей. Возьмем матрицу смежности графа G. Заменим все элементы этой матрицы равные 0 на символ λ, диагональные элементы матрицы положим равными 0. Полученная матрица будет лежать в классе D, обозначим её через R. Рассмотрим i-ую и j-ую строки матрицы R, $i \neq j$. Все координаты строк, кроме i-ой и j-ой, равны либо 1, либо λ. i-ая координата i-ой строки равна 0, и j-ая координата j-ой строки равна 0. Если i-ая и j-ая вершины графа G смежны, то i-ая координата j-ой строки равна 1, j-ая координата i-ой строки равна 1, и i-ая и j-ая строки матрицы R не являются смежными. Если i-ая и j-ая вершины графа G не являются смежными, то i-ая координата j-ой строки равна λ, j-ая координата i-ой строки равна λ, и i-ая и j-ая строки матрицы R являются смежными. Таким образом граф G является графом отделимости матрицы R. По утверждению 2 раскраска вершин графа G в k цветов существует тогда и только тогда, когда существует матрица пользователей U, порождающая матрицу R, с числом типов пользователей не больше k. $\square$

Следствие 4. *Пусть R — матрица рейтингов, M — натуральное число, $M > 2$. Задача определения, существует ли представитель матриц пользователей при ограничении на число типов пользователей для пары (R, M), является NP-полной.*

Следствие 5. *Пусть R — матрица рейтингов, M — натуральное число, $M > 2$. Задача определения, существует ли представитель матриц пользователей с заданным числом типов пользователей для пары (R, M), является NP-полной.*

Таким образом задача поиска представителя матриц пользователей при ограничении на число типов пользователей для пары (R, M) при $M > 2$, задача поиска представителя матриц пользователей с заданным числом типов пользователей для пары (R, M) при $M > 2$ и задача поиска представителя матриц пользователей с минимальным числом типов пользователей являются NP-трудными.

Алгоритм поиска представителя матриц пользователей при ограничении на число типов пользователей для пары (R, M):

1) Строим граф отличимости G матрицы R.
2) Находим любую раскраску вершин графа G в M цветов. Если такой раскраски не существует, то не существует представителя матриц пользователей, конец алгоритма.
3) Строки матрицы R, соответствующие вершинам графа G одного цвета, доопределяем до одной и той же строки над алфавитом A. Полученная матрица является представителем матриц пользователей при ограничении на число типов пользователей для пары (R, M). Конец алгоритма.

Алгоритм поиска представителя матриц пользователей с минимальным числом типов пользователей:

1) Строим граф отличимости G матрицы R.
2) Находим хроматическое число $\chi(G)$ графа G.
3) Находим любую раскраску вершин графа G в $\chi(G)$ цветов.
4) Строки матрицы R, соответствующие вершинам графа G одного цвета, доопределяем до одной и той же строки над алфавитом A. Полученная матрица является представителем матриц пользователей с минимальным числом типов пользователей. Конец алгоритма.

Алгоритм полиномиальной сложности поиска представителя матриц пользователей при ограничении на число типов пользователей для пары $(R, 2)$

Рассмотрим случай поиска представителя матриц пользователей при ограничении на число типов пользователей для пары $(R, 2)$. В этом случае существует полиномиальный алгоритм поиска представителя матриц пользователей.

Алгоритм поиска представителя матриц пользователей при ограничении на число типов пользователей для пары $(R, 2)$:

1) Строим граф отличимости G матрицы R.

2) Проверяем граф G на двудольность. Если граф является двудольным, то находим его доли. Иначе не существует представителя матриц пользователей, конец алгоритма.

3) Строки матрицы R, соответствующие вершинам графа G из одной доли, доопределяем до одной и той же строки над алфавитом A. Полученная матрица является представителем матрицы пользователей для пары $(R, 2)$. Конец алгоритма.

Пусть матрица R состоит из m строк и n столбцов. Граф отличимости матрицы R можно задать матрицей смежности. Для построения матрицы смежности графа отличимости в худшем случае понадобится $O(m^2 n)$ операций. Пространственная сложность равна $O(m^2)$. Чтобы проверить граф на предмет двудольности, достаточно в каждой компоненте связности выбрать любую вершину и помечать оставшиеся вершины во время обхода графа (например, поиском в ширину[42]) поочередно как четные и нечетные. Временная сложность поиска в ширину составляет $O(m^2)$. Пространственная сложность — $O(m^2)$. Для доопределения строк матрицы R в худшем случае понадобится $O(mn)$ операций. Тем самым временная сложность алгоритма поиска представителя матриц пользователей для пары $(R, 2)$ равна $O(m^2 n)$, пространственная сложность — $O(m^2)$.

Перечисление матриц пользователей

Пусть R — матрица рейтингов. Задачу перечисления всех матриц пользователей U, которые порождает матрицу R, будем называть *задачей перечисления матриц пользователей для матрицы рейтингов R*.

Задачу перечисления матриц пользователей U с числом типов пользователей не больше чем M, которые порождают матрицу R, будем называть *задачей перечисления матриц пользователей при ограничении на число типов пользователей для пары (R, M)*.

Задачу перечисления матриц пользователей U с числом типов пользователей равным M, которые порождают матрицу R, будем называть *задачей перечисления матриц пользователей с заданным числом типов пользователей для пары (R, M)*.

Задачу перечисления матриц пользователей U с минимальным числом типов пользователей, которые порождают матрицу R, будем называть *задачей перечисления матриц пользователей с минимальным числом типов пользователей для матрицы рейтингов R*.

Задачи перечисления матриц пользователей являются более сложными задачами, чем задачи поиска представителя матриц пользователей. Поэтому задача перечисления матриц пользователей при ограничении на число типов пользователей для пары (R, M) при $M > 2$, задача перечисления матриц пользователей с заданным числом типов пользователей для пары (R, M) при $M > 2$ и задача перечисления матриц пользователей с минимальным числом типов пользователей являются NP-трудными.

Алгоритм перечисления матриц пользователей при ограничении на число типов пользователей для пары (R, M):

1) Строим граф отличимости G матрицы R.

2) Находим все раскраски вершин графа G в M цветов. Если раскрасок вершин графа не существует, то матриц пользователей для данной пары (R, M) не существует, конец алгоритма.

3) Для каждой раскраски вершин графа G строим матрицу P, состоящую из элементов алфавита A и переменных $\alpha_1, \alpha_2, \ldots, \alpha_s$. Матрица P будет задавать множество матриц пользователей, где вместо переменных $\alpha_1, \alpha_2, \ldots, \alpha_s$ могут стоять любые элементы из алфавита A. Множество матриц P будет задавать перечисление матриц пользователей при ограничении на число типов пользователей для пары (R, M).

   Построение матрицы P.

   а) Для заданной раскраски вершин графа G для каждого множества вершин, окрашенных в один цвет, и тем самым соответствующих системе попарно смежных строк матрицы R, строим главный порождающий вектор в соответствии с алгоритмом, описанным в доказательстве утверждения 1.

   б) В системе главных порождающих векторов заменяем все элементы λ на переменные $\alpha_1, \alpha_2, \ldots, \alpha_s$ так, чтобы в системе главных порождающих векторов не было одинаковых переменных.

в) Строим матрицу P из матрицы R, заменяя каждую строку матрицы R на соответствующий ей главный порождающий вектор.

Конец алгоритма.

Алгоритм перечисления матриц пользователей с минимальным числом типов пользователей состоит в том, что мы сначала находим хроматическое число $\chi(G)$ графа отличимости G матрицы R, а затем применяем предыдущий алгоритм.

Список литературы

[1] В.Б.Кудрявцев, С.В. Алешин, А.С. Подколзин. Введение в теорию автоматов. Издательство «Наука», Москва, 1985.

[2] Пантелеев П.А. Об отличимости состояний решетчатых автоматов // Интеллектуальные системы. — 2004. — Т. 8, вып. 1–4. — С. 529–542.

[3] Кирнасов А.Е. Об отношении сложностей условного и безусловного установочного экспериментов // Интеллектуальные системы. — 2005. — Т. 9, вып. 1–4. — С. 433–444.

[4] Уваров Д.В. О сложности кратных диагностических экспериментов для подмножеств состояний автоматов // Интеллектуальные системы. — 2005. — Т. 9, вып. 1–4. — С. 485–504.

[5] Кудрявцев В.Б., Грунский И.С., Козловский В.А. Анализ и синтез автоматов по их поведению // Интеллектуальные системы. — 2006. — Т. 10, вып. 1–4. — С. 345–448.

[6] Пантелеев П.А. Об отличимости состояний автомата при искажениях на входе // Интеллектуальные системы. — 2007. — Т. 11, вып. 1–4. — С. 653–678.

[7] Гасанов Э.Э. Прогнозирование периодических сверхсобытий автоматами // Интеллектуальные системы. — 2015. — Т. 19, вып. 1–4. — С. 23–34.

[8] Ансель Ж. О числе монотонных булевых функций n переменных. — Кибернетический сборник, новая серия, вып. 5, 1968, С. 53–57.

[9] Angluin D. Queries and Concept Learning //Machine Learning. Vol. 2. 1988. P. 319–342.

[10] Осокин В.В. Асимптотически оптимальный алгоритм расшифровки разбиения булевого куба на подкубы // Интеллектуальные системы. — 2007. — Т. 11, вып. 1–4. — С. 635–652.

[11] Воронин Б.В., Осокин В.В. О сложности расшифровки существенных переменных функции, задающей разбиение булевого куба // Интеллектуальные системы. — 2008. — Т. 12, вып. 1–4. — С. 159–178.

[12] Осокин В.В. О параллельной расшифровке разбиений булевого куба // Интеллектуальные системы. — 2009. — Т. 13,вып. 1–4. — С. 427–454.

[13] Осокин В.В. О параллельной параметро-эффективной расшифровке псевдо-булевских функций // Интеллектуальные системы. — 2010. — Т. 14,вып. 1–4. — С. 429–458.

[14] Хегай С.И. Расшифровка полиномиальных функций ранжирования // Интеллектуальные системы. 2015. 19:1. 213 – 230.

[15] Valiant, L. G. A theory of the learnable. ACM Press New York, NY, USA, 1984, Volume 27, Issue II, 1134-1142.

[16] Blum A. Learning a Function of r Relevant Variables. COLT 2003, Open problems.

[17] Arpe J., Reischuk B. Learning Juntas in the Presence of Noise. Theoret. Comput. Sci. 384(1): 2-21, 2007.

[18] Гончаров М. Системы выработки рекомендаций. [http://www.businessdataanalytics.ru/RecommendationSystems.htm].

[19] Ricci F., Rokach L., Shapira B., Kantor P.B. Recommender Systems Handbook. — Springer, 2011.

[20] Жернакова О. Системы рекомендаций и поиска видеоконтента // Интернет-журнал по широкополосным сетям и мультимедийным технологиям "Телемультимедиа". — 2012.

[21] Mooney R.J., Roy L. Content-based book recommendation using learning for text categorization // In Workshop Recom. Sys.: Algo. and Evaluation. — 1999.

[22] Melville P.,Mooney R., Nagarajan R. Content-Boosted Collaborative Filtering for Improved Recommendations // University of Texas, USA. — 2002. — P. 187–192.

[23] Felfernig A., Burke R. Constraint-based Recommender Systems: Technologies and Research Issues // Proceedings of the ACM International Conference on Electronic Commerce (ICEC'08). — 2008. — P. 17–26.

[24] Su X., Khoshgoftaar T.M. A Survey of Collaborative Filtering Techniques A Survey of Collaborative Filtering Techniques // Hindawi Publishing Corporation, Advances in Artificial Intelligence archive. — 2009. — P. 1–19.

[25] Linden G., Smith B., York J. Item-to-Item Collaborative Filtering // IEEE Internet Computing, Los Alamitos. — 2003. — P. 76–80.

[26] Sarwar B., Karypis G., Konstan J., and Riedl J. Item-Based Collaborative Filtering Recommendation Algorithms // University of Minnesota, Minneapolis. — 2001. — P. 285–295.

[27] Goldberg K., Roeder T., Gupta D., Perkins C. Eigentaste: A Constant Time Collaborative Filtering Algorithm // IEOR and EECS Departments University of California. — 2000.

[28] Saric A., Hadzikadic M., Wilson D. Alternative Formulas for Rating Prediction Using Collaborative Filtering // College of Computing and Informatics.

[29] Burke R. Hybrid Web Recommender Systems // The Adaptive Web. — 2007. — P. 377–408.

[30] Бессалов А.С., Рыжов А.П. Решение задачи распознавания блокируемых объявлений с помощью методов обработки естественных языков // Интеллектуальные системы. — 2014. — Т. 18, вып. 1–4. — С. 5–14.

[31] Александров Д.Е. Об оценках автоматной сложности распознавания класса регулярных языков // Интеллектуальные системы. — 2014. — Т. 18, вып. 1–4. — С. 161–190.

[32] Агниашвили П.Г. Алгоритм восстановления изображения по его коду // Интеллектуальные системы. — 2014. — Т. 18, вып. 1–4. — С. 183–204. .

[33] Pandora Media, Inc. [http://www.pandora.com].

[34] Last.fm Ltd. [http://www.lastfm.ru/].

[35] Candes E.J., Recht B. Exact Matrix Completion via Convex Optimization // Applied and Computational Mathematics, Caltech, Pasadena, CA 91125. Center for the Mathematics of Information, Caltech, Pasadena, CA 91125. — 2008.

[36] Boumal N., Absil P.-A. RTRMC: A Riemannian trust-region method for low-rank matrix completion.

[37] Abernethy J., Bach F., Evgeniou T., Vert J.-P. A New Approach to Collaborative Filtering: Operator Estimation with Spectral Regularization // Journal of Machine Learning Research. — 2009. — P. 803–826.

[38] Mazumder R., Hastie T., Tibshirani R. Spectral Regularization Algorithms for Learning Large Incomplete Matrices // Journal of Machine Learning Research. — 2010. — P. 2287–2322.

[39] Cai I.-F., Candes E.J., Shen Z. A singular value thresholding algorithm for matrix completion // SIAM. — 2010. — P. 1956–1982.

[40] Кабанова А.С. О новом подходе к задаче матричного пополнения // Научно-практическая конференция "Старт в науку". — 2012.

[41] Хлебников М.В., Щербаков П.С. Подходы к решению задачи матричного устойчивого пополнения // Автоматика. — 2011. — С. 259.

[42] Емеличев В.А., Мельников О.И., Сарванов В.И., Тышкевич Р.И. Лекции по теории графов. — Наука, 1990.

The complexity of ratings matrices completion problem for recommendation systems

A. O. Savinskii

A mathematical model of recommendation systems is constructed. Estimates of users types number in matrix of users is found. It is found out, that problems of search matrix of users and enumeration of them in general case are a NP-hardness problems. Furthermore, polynomial algorithm for search matrix of users when the user types number is two is provided.

Keywords: recommendation systems, matrix completion, search matrix of users, enumeration of matrices of users