3. Определение минимальной ширины канала между парой компонентов при топологической трассировке / А. В. Бессонов, К. А. Кноп, Ю. Т. Лячек, Ю. И. Попов // Изв. СПбГЭТУ «ЛЭТИ». 2013. № 10. С. 31–34.

4. Бессонов А. В., Кноп К. А., Лячек Ю. Т. Декомпозиция задачи размещения компонентов // Изв. СПбГЭТУ «ЛЭТИ». 2014. № 1. С. 11–15.

A. V. Bessonov, S. Yu. Luzin
Ltd. «Eremex» (Saint-Petersburg)

Yu. T. Lyachek
Saint-Petersburg state electrotechnical university «LETI»

DEFINITION OF NEIGHBORHOODS MULTIPOLE

A method is proposed for local optimization of component placement. The method is based on the determination two-pole components in the neighborhood of a multiterminal and creation a united group –"a virtual multiterminal component". This approach reduces the overall size of the location problem and simplify the process of placing the components of the group to the required resource tracing connections.

Printed circuit board, autoplacement, virtual multiterminal component

УДК: 20.53.19, 28.23.13

Е. Г. Воробьев
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Сжатие двоичных кодов на основе традиционных методов и использования псевдорегулярных чисел

Проведен сравнительный анализ существующих методов сжатия и нового, основанного на использовании чисел с псевдорегулярной двоичной структурой. Предложен подход, позволяющий решить проблему хранения резервной информации больших объемов, что характерно для облачных структур и кластерных систем центров обработки данных.

Сжатие информации, псевдорегулярная двоичная структура, методы и алгоритмы сжатия, уменьшение воздействия на объем хранимой информации

С каждым годом растет объем хранимой, передаваемой и обрабатываемой информации в информационных системах. При этом основной вклад вносят данные, в то время как системное и прикладное программное обеспечение меняется очень медленно.

В связи с высокой компьютеризацией общественной жизни возникает проблема сохранения больших объемов используемой и резервной информации, а также передачи ее потребителю, причем особенностью современного периода является то, что информация, накопленная в базах данных центральных серверов министерств и ведомств, насчитывает сотни терабайт и эта цифра постоянно растет каждый день. При этом воздействие на информационные системы антропогенных, техногенных и стихийных факторов приводит к потерям, которые могут сделать невозможной любую целенаправленную деятельность с применением указанной вычислительной техники.

Концепция хранения, обработки данных в условиях недостаточных объемов памяти вычислительных средств и передачи их в условиях недостаточной пропускной способности базируется на методах и алгоритмах сжатия информации. Сжатие данных – это процедура перекодирования данных, позволяющая уменьшить их объем, которая применяется для более рационального использования устройств хранения и передачи данных.

Проблема совершенствования методов и алгоритмов сжатия актуальна всегда, но ее решению мешают некоторые положения теории информации, которые считаются непреложными и создают известные ограничения для развития.

Ограничения современных методов и алгоритмов сжатия информации. Далее рассмотрим известные основные положения теории сжатия информации. Наиболее существенные с точки зрения дальнейшего обсуждения положения выделены автором курсивом и снабжены комментариями.

Ограничение 1. По современным подходам, в основе любого способа сжатия информации лежат несколько моделей, первая и наиболее важная из которых – модель источника информации, или, более конкретно, модель избыточности. Иными словами, для сжатия информации используются некоторые сведения о том, какого рода информация сжимается (считается, что, не обладая никакими сведениями об информации, нельзя сделать ровным счетом никаких предположений, какое преобразование позволит уменьшить объем сообщения). Во всех случаях это частота встречаемости в сообщении символов или комбинаций символов. Данная информация используется в процессе сжатия и разжатия. Модель избыточности может также строиться или параметризоваться на этапе сжатия. Методы, позволяющие на основе входных данных изменять модель избыточности информации, называются адаптивными. Неадаптивными являются обычно узкоспецифичные алгоритмы, применяемые для работы с хорошо определенными и неизменными характеристиками. Подавляющая же часть достаточно универсальных алгоритмов являются в той или иной мере адаптивными.

Любой метод сжатия информации включает в себя 2 преобразования, обратных друг другу:

– преобразование сжатия (компрессии);

– преобразование расжатия (декомпрессии).

Преобразование сжатия обеспечивает получение сжатого сообщения из исходного. Декомпрессия же обеспечивает получение исходного сообщения (или его приближения) из сжатого.

Все методы сжатия делятся на 2 основных класса: без потерь и с потерями. Кардинальное различие между ними в том, что сжатие без потерь позволяет точно восстановить исходное сообщение, а сжатие с потерями – получить лишь некоторое приближение исходного сообщения, т. е. отличающееся от исходного, но в пределах некоторых заранее определенных погрешностей.

Эти погрешности должны определяться другой моделью – моделью приемника, устанавливающей, какие данные и с какой точностью представления важны для получателя, а какие допустимо не учитывать.

Основными техническими характеристиками процессов сжатия и результатов их работы являются:

– *степень сжатия* (compress rating), или отношение (ratio) объемов исходного и результирующего потоков (коэффициент сжатия);

– *скорость сжатия* – время, затрачиваемое на сжатие некоторого объема информации входного потока до получения из него эквивалентного выходного потока;

– *качество сжатия* – величина, показывающая, насколько сильно упакован выходной поток, в результате повторного его сжатия по этому же или иному алгоритму.

Ограничение 2. Коэффициент сжатия – основная характеристика алгоритма сжатия, выражающая основное прикладное качество. Она определяется как отношение размера несжатых данных к сжатым, т. е. $k = S_o/S_c$, где k – коэффициент сжатия; S_o – размер несжатых данных; S_c – размер сжатых.

Таким образом, чем выше коэффициент сжатия, тем алгоритм лучше (хотя размер сжатых данных в условиях ограниченности размеров памяти носителя информации является более существенным показателем эффективности – *прим. автора*).

Далее теория гласит:

– если $k = 1$, то сжатие не производится и размер выходного сообщения равен входному;

– если $k < 1$, то алгоритм порождает при сжатии сообщение большего размера, нежели несжатое, т. е. совершает «вредную» работу. Ситуация с $k < 1$ вполне возможна при сжатии.

Коэффициент сжатия может быть как постоянным (некоторые алгоритмы сжатия звука, изображения и т. п., например A-закон, μ-закон, ADPCM), так и переменным. Во втором случае он может быть определен либо для какого-либо конкретного сообщения либо оценен по некоторым критериям:

– среднее (обычно по некоторому тестовому набору данных);

– максимальное (случай наилучшего сжатия);

– минимальное (случай наихудшего сжатия) и др.

Ограничение 3. Невозможно получить алгоритм сжатия без потерь, который при любых данных образовывал бы на выходе данные меньшей или равной длины.

Обоснование этого факта заключается в том, что количество различных сообщений длиной n составляет ровно 2^n. Тогда количество различных сообщений с длиной меньшей или равной n (при наличии хотя бы одного сообщения меньшей длины) будет меньше 2^n. Это значит, что невозможно однозначно сопоставить все исходные сообщения сжатым: либо некоторые исходные сообщения не будут иметь сжатого представления, либо нескольким исходным сообщениям будет соответствовать одно и то же сжатое, а значит, их нельзя различить.

Современные методы и алгоритмы сжатия.

Как уже отмечалось, сжатие бывает без потерь (когда возможно восстановление исходных данных без искажений) или с потерями (восстановление возможно с искажениями, не существенными с точки зрения дальнейшего использования восстановленных данных). Сжатие без потерь обычно используется при обработке компьютерных программ и данных, реже – для сокращения объема звуковой, фото- и видеоинформации. Сжатие с потерями применяется для сокращения объема звуковой, фото- и видеоинформации, оно значительно эффективнее сжатия без потерь.

Сжатие данных с потерями – это метод сжатия данных, при котором распакованный файл отличается от оригинального, но «достаточно близок» для того, чтобы быть полезным каким-то образом. Этот тип компрессии часто используется в Интернете, особенно в потоковой передаче данных и телефонии. В данном контексте эти методы часто называются кодеками.

Существует две основные схемы сжатия с потерями:

– в трансформирующих кодеках фреймы изображений или звука трансформируются в новое базисное пространство и производится квантование. Трансформация может осуществляться либо для всего фрейма целиком (как, например, в схемах на основе вейвлет-преобразования), либо поблочно (характерный пример – JPEG). Результат затем сжимается энтропийными методами;

– в предсказывающих кодеках предыдущие и/или последующие данные используются для того, чтобы предсказать текущую выборку («сэмпл») изображения или звука. Ошибка между предсказанными и реальными данными вместе с добавочной информацией, необходимой для предсказания, затем квантуется и кодируется.

В некоторых системах эти две техники комбинируются использованием трансформирующих кодеков для сжатия ошибочных сигналов, сгенерированных на стадии предсказания.

Преимущество методов сжатия с потерями над методами сжатия без потерь состоит в том, что при существенном превосходстве в степени сжатия они продолжают удовлетворять поставленным требованиям, а именно – искажения должны быть в допустимых пределах чувствительности человеческих органов.

Методы сжатия с потерями часто используются для сжатия аналоговых данных – чаще всего звука или изображений.

В таких случаях распакованный файл может очень сильно отличаться от оригинала на уровне сравнения «бит в бит», но практически не отличим для человеческого уха или глаза в большинстве реальных применений.

Коэффициент сжатия с потерями при этом сильно зависит от допустимой погрешности сжатия или его *качества*, которое обычно выступает как параметр алгоритма.

Основным критерием различия между алгоритмами сжатия является описанное ранее наличие или отсутствие потерь. В общем случае алгоритмы сжатия без потерь универсальны в том смысле, что их можно использовать на данных любого типа, в то время как применение сжатия с потерями должно быть обосновано. Некоторые виды данных не приемлют каких бы то ни было потерь:

– символические данные, изменение которых неминуемо приводит к изменению их семантики: программы и их исходные тексты, двоичные массивы и т. п.;

– жизненно важные данные, изменения в которых могут привести к критическим ошибкам, например данные, получаемые с медицинской измерительной техники или контрольных приборов летательных, космических аппаратов и т. п.;

– данные, многократно подвергаемые сжатию и декомпрессии: рабочие графические, звуковые, видеофайлы.

Однако сжатие с потерями позволяет добиться гораздо больших коэффициентов сжатия за счет отбрасывания незначащей информации, которая плохо сжимается. Так, например, алгоритм сжатия звука без потерь FLAC позволяет в большинстве случаев сжать звук в 1.5–2.5 раза, в то

время как алгоритм с потерями Vorbis (в зависимости от установленного параметра качества) – даже в 15 раз с сохранением приемлемого качества звучания.

Сжатие без потерь (англ. Lossless data compression) – метод сжатия информации, при котором закодированная информация может быть восстановлена с точностью до бита. При этом оригинальные данные полностью восстанавливаются из сжатого состояния. Этот тип сжатия принципиально отличается от сжатия данных с потерями. Для каждого из типов цифровой информации, как правило, существуют свои оптимальные алгоритмы сжатия без потерь.

Сжатие данных без потерь используется во многих приложениях, например во всех файловых архиваторах; как компонент в сжатии с потерями. Сжатие без потерь применяется, когда важна идентичность сжатых данных оригиналу. Обычный пример – исполняемые файлы и исходный код. Некоторые графические файловые форматы, такие, как PNG или GIF, используют только сжатие без потерь, тогда как другие (TIFF, MNG) могут использовать сжатие как с потерями, так и без.

Техника сжатия без потерь базируется на следующей теореме (приведем без доказательства): для любого N нет алгоритма сжатия без потерь, который:

1) любой файл длиной не более N байт или оставляет той же длины, или уменьшает;

2) существует файл длиной не более N, который уменьшается хотя бы на 1 байт.

Впрочем, реальные данные обычно имеют высокую информационную энтропию (меру хаотичности информации, неопределенность появления какого-либо символа первичного алфавита, которая при отсутствии информационных потерь численно равна количеству информации на символ передаваемого сообщения), а любой алгоритм сжатия можно модифицировать так, чтобы он увеличивал размер не более чем на 1 бит. К тому же за счет специализации алгоритмов под некоторый тип данных (текст, графику, звук и т. д.) удается добиться высокой степени сжатия: так, применяющиеся в архиваторах универсальные алгоритмы сжимают звук примерно на треть (в 1.5 раза), в то время как алгоритм FLAC – в 2.5 раза.

Большинство специализированных алгоритмов малопригодны для файлов других типов: например, звуковые данные плохо сжимаются алгоритмом, рассчитанным на тексты.

Большинство алгоритмов сжатия без потерь работают в две стадии:

– генерация статистической модели для входящих данных;

– отображение входящих данных в битовом представлении с использованием модели для получения «вероятностных» (т. е. часто встречаемых) данных, которые используются чаще, чем «невероятностные».

Системные требования алгоритмов. Различным алгоритмам для своего исполнения необходимо различное количество ресурсов вычислительной системы: оперативной памяти (под промежуточные данные); постоянной памяти (под код программы и константы); процессорного времени.

В целом эти требования зависят от сложности и «интеллектуальности» алгоритма. По общей тенденции, чем лучше и универсальнее алгоритм, тем большие требования к машине он предъявляет. Однако в специфических случаях простые и компактные алгоритмы могут работать лучше. Системными требованиями определяются их потребительские качества: чем менее требователен алгоритм, тем на более простой, а следовательно, компактной, надежной и дешевой системе он может работать.

Так как алгоритмы сжатия и разжатия работают в паре, то имеет значение также соотношение системных требований к ним. Нередко, усложнив один алгоритм, можно значительно упростить другой. Таким образом, возможны 3 варианта:

1. Алгоритм сжатия гораздо требовательнее к ресурсам, нежели алгоритм разжатия.

2. Алгоритмы сжатия и разжатия имеют примерно равные требования.

3. Алгоритм сжатия существенно менее требователен, чем алгоритм разжатия.

Примеры форматов и их реализаций:

– универсальные – Zip, 7-Zip, RAR, GZip, PAQ и др.;

– звук – FLAC (Free Lossless Audio Codec), Monkey's Audio (APE), TTA (True Audio), TTE, LA (LosslessAudio), RealAudio Lossless, WavPack и др.;

– изображения – BMP, PNG;

– видео – Huffyuv;

– текст – HA.

Рассмотрим истоки ограничений для реализации методов сжатия. Для этого будем проверять основные положения теории информации и попытаемся определить, какие из них подлежат пере-

смотру. Особое внимание уделим методу сжатия без потерь, так как сжатие с потерями появилось всего лишь как необходимый компромисс.

Итак, теория информации утверждает: «Сжатие основано на устранении избыточности информации, содержащейся в исходных данных. Примером избыточности является повторение в тексте фрагментов (например, слов естественного или машинного языка). Подобная избыточность обычно устраняется заменой повторяющейся последовательности более коротким значением (кодом). Другой вид избыточности связан с тем, что некоторые значения в сжимаемых данных встречаются чаще других, при этом возможно заменять часто встречающиеся данные более короткими кодами, а редкие – более длинными (вероятностное сжатие). Сжатие данных, не обладающих свойством избыточности (например, случайный сигнал или шум, зашифрованная информация), невозможно без потерь».

Энтропийное кодирование – кодирование последовательности значений с возможностью однозначного восстановления с целью уменьшения объема информации (длины последовательности) с помощью усреднения вероятностей появления элементов последовательности. Академик А. Н. Колмогоров определил сложность двоичной строки как длину кратчайшей программы для универсального компьютера, способной генерировать эту строку. Такое представление о сложности независимо и почти одновременно предложили Г. Чайтин и Р. Соломонофф.

Предполагается, что до кодирования отдельные элементы последовательности имеют различную вероятность появления. После кодирования в результирующей последовательности вероятности появления отдельных символов практически одинаковы (энтропия на символ максимальна).

Различают несколько вариантов кодов:

– сопоставление каждому элементу исходной последовательности различного числа элементов результирующей последовательности. Чем больше вероятность появления исходного элемента, тем короче соответствующая результирующая последовательность (примеры – код Шеннона–Фано, код Хаффмана);

– сопоставление нескольким элементам исходной последовательности фиксированного числа элементов конечной последовательности (пример – код Танстола);

– другие структурные коды, основанные на операциях с последовательностью символов (пример – кодирование длин серий);

– если приблизительные характеристики энтропии потока данных предварительно известны, может быть полезен более простой статический код, такой, как унарное кодирование, гамма-код Элиаса, кодирование Фибоначчи, кодирование Голомба или кодирование Райса.

Наиболее важным является следующее заявление теории: «согласно теореме Шеннона, существует предел сжатия без потерь, зависящий от энтропии источника. Чем более предсказуема получаемая информация, тем лучше ее можно сжать. Случайная последовательность сжатию без потерь не поддается».

По сути дела все недостатки существующих систем сжатия базируются на положениях двух теорем К. Шеннона, известных как прямая и обратная.

В применении к побуквенному кодированию прямая теорема может быть сформулирована следующим образом [1]: «Существует префиксный, т. е. разделимый код, для которого средняя длина сообщений отличается от нормированной энтропии не более, чем на единицу».

В качестве доказательства теоремы исследуются характеристики кода Шеннона–Фано. Данный код удовлетворяет условиям теоремы и обладает указанными свойствами.

Алгоритм Шеннона–Фано – один из первых алгоритмов сжатия, который впервые сформулировали американские ученые. Данный метод во многом сходен с алгоритмом Хаффмана, который появился на несколько лет позже. Алгоритм использует коды переменной длины: часто встречающийся символ кодируется кодом меньшей длины, редко встречающийся – кодом большей длины. Коды Шеннона–Фано префиксные, т. е. никакое кодовое слово не является префиксом любого другого. Это свойство позволяет однозначно декодировать любую последовательность кодовых слов.

Обратная теорема ограничивает максимальную степень сжатия, достижимую с помощью кодирования без потерь. Применительно к побуквенному кодированию она описывает ограничение на среднюю длину кодового слова для любого *разделимого* кода. Для любого разделимого кода средняя длина сообщений больше или равна энтропии источника, нормированной на двоичный логарифм от числа букв в алфавите кодера.

Важно то, что данные ограничения имели смысл только для указанного алгоритма и служили для доказательства его эффективности.

Существующие методы сжатия	Метод с использованием ПРЧ
В основе любого способа сжатия информации лежат несколько моделей, первая и наиболее важная из которых – модель источника информации, или, более конкретно, модель избыточности	Не требует анализа входной информации для построения модели избыточности, выявления повторяющихся элементов и т. д.
Не обладая никакими сведениями об информации, нельзя сделать ровным счетом никаких предположений, какое преобразование позволит уменьшить объем сообщения. Во всех случаях это частота встречаемости в сообщении символов или комбинаций символов	Использование симметричных относительно ПРЧ отображений в двоичных полях последовательно уменьшает количество значащих разрядов, т. е. приводит к уменьшению объема сообщения до значения, установленного оператором системы
Допустимы погрешности восстановления данных при использовании метода сжатия с потерями. Эти погрешности должны определяться другой моделью – моделью приемника, определяющей, какие данные и с какой точностью представления важны для получателя, а какие допустимо не учитывать	Важны все данные в полном объеме. В силу этого имеют право на существование только методы сжатия без потерь. В остальных случаях – это отсутствие математических методов работы с длинными двоичными кодами. Данный метод свободен от этого недостатка
Коэффициент сжатия – основная характеристика алгоритма сжатия, выражающая основное прикладное качество. Она определяется как отношение размера несжатых данных к сжатым. Таким образом, чем выше коэффициент сжатия, тем алгоритм лучше	Лучше алгоритм, который позволяет оператору указать допустимый объем сжатой информации при произвольном объеме исходной. Данное требование выполняется только этим методом
Невозможно получить алгоритм сжатия без потерь, который при любых данных образовывал бы на выходе данные меньшей или равной длины	Данный метод позволяет не учитывать формат представления информации
Сжатие с потерями позволяет добиться гораздо больших коэффициентов сжатия, чем другие методы, за счет отбрасывания незначащей информации, которая плохо сжимается	Необходимо различать остаточный объем сжатых данных и объем возникающей при этом служебной информации. Метод хорошо сжимает любую информацию, так как работает на уровне машинных кодов
Для каждого из типов цифровой информации, как правило, существуют свои оптимальные алгоритмы сжатия без потерь. Большинство специализированных алгоритмов малопригодны для файлов других типов: например, звуковые данные плохо сжимаются алгоритмом, рассчитанным на тексты	Метод оптимален для всех форматов файлов, так как работает на уровне машинных кодов, обладает меньшей ресурсоемкостью за счет использования теории комплексных чисел при реализации метода представления информации и реализации алгоритма вместо физического представления информации в памяти ЭВМ
Чем лучше и универсальнее алгоритм, тем большие требования к ЭВМ он предъявляет	Расход ресурсов ЭВМ постоянен и не зависит от особенностей данного алгоритма
Сжатие данных, не обладающих свойством избыточности (например, случайный сигнал или шум, зашифрованная информация), невозможно без потерь	Позволяет реализовывать сжатие без потерь для любых типов файлов
Согласно теореме Шеннона, существует предел сжатия без потерь, зависящий от энтропии источника. Чем более предсказуема получаемая информация, тем лучше ее можно сжать. Случайная последовательность сжатию без потерь не поддается	Нижний предел сжатия данных для данного метода равен 0, верхний определяется оператором. При этом объем служебной информации остается постоянным, но зависит от заданного числа шагов алгоритма

Как бы в противоположность заявлению Шеннона о том, что случайная последовательность сжатию без потерь не поддается, были разработаны алгоритмы сжатия текстов/файлов неизвестного формата.

Имеется 2 основных подхода к сжатию файлов неизвестного формата [2], [3]:

- На каждом шаге алгоритма сжатия либо следующий символ помещается как есть (со специальным флагом, помечающим, что он не сжат), либо указываются границы слова из предыдущего куска, которое совпадает со следующими символами файла. Разархивирование сжатых таким образом файлов выполняется очень быстро, поэтому данные алгоритмы используются для создания самораспаковывающихся программ.
- Для каждой последовательности в каждый момент времени собирается статистика ее встречаемости в файле. На основе этой статистики вычисляется вероятность значений для очередного символа. После этого можно применять ту или иную разновидность статистического кодирования, например арифметическое кодирование или кодирование Хаффмана, для замены часто встречающихся последовательностей на более короткие, а редко встречающихся – на более длинные.

Таким образом, задача снова сводится к известным подходам, основанным на модели избыточности.

В связи с этим сравним основные характеристики существующих методов сжатия и метода, основанного на применении псевдорегулярных чисел (ПРЧ), описанного в работе автора [4] (см. таблицу).

Метод сжатия с использованием псевдорегулярных чисел является универсальным и может служить основой для построения систем резервного копирования информации в глобальных информационных системах, а также систем обмена информационными кластерами. При переходе к квантовым технологиям метод позволяет создать хранилища данных (storage) с новой технологией представления информации.

СПИСОК ЛИТЕРАТУРЫ

1. Габидулин Э. М., Пилипчук Н. И. Теоремы Шеннона для источника // Лекции по теории информации. М.: Изд-во МФТИ, 2007. С. 49–52.

2. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео / Д. А. Ватолин, А. В. Ратушняк, М. В. Смирнов, В. А. Юкин. М.: Диалог-МИФИ, 2002. С. 384.

3. Сэломон Д. Ю. Сжатие данных, изображения и звука. М.: Техносфера, 2004. С. 368.

4. Воробьев Е. Г., Цехановский В. В. Псевдорегулярные числа в двоичных полях // Изв. СПбГЭТУ «ЛЭТИ». 2014. № 2. С. 18–22.

E. G. Vorobiev
Saint-Petersburg state electrotechnical university «LETI»

COMPRESSION OF BINARY CODES ON THE BASIS OF TRADITIONAL METHODS AND USE OF PSEUDO-REGULAR NUMBERS

In article the comparative analysis of the existing methods of data compression and new, on the basis of use of numbers with pseudo-regular binary structure is carried out. The approach allows to solve a problem of storage of reserve information of large volumes that is characteristic for cloudy structures and cluster systems of data-processing centers are offered.

Data compression, pseudo-regular binary structure, methods and algorithms of compression, reduction of impact on volume of the stored information

УДК 681.3, 004.031.4

М. М. Заславский, Т. А. Берленко
ООО «Fruct» (Санкт-Петербург)

Реализация механизма подбора рекомендаций в информационной системе «Открытая Карелия»

Рассмотрен гибкий подход к построению рекомендаций на основании баллов близости для информационной системы с анонимным доступом без использования информации о пользователе и его оценках содержимого этой системы. Даны определения основных понятий подхода. Приведены примеры формул для вычисления баллов близости при сравнении содержимого по данным различной природы (полнотекстовые поля, теги, поля с конечным множеством значений). Описана программная реализация системы подбора рекомендаций для ИС «Открытая Карелия», приведены ее ограничения и направления для ее дальнейшего улучшения.

Рекомендательные системы, баллы близости, системы с анонимным доступом

На сегодняшний день системы подбора рекомендаций стали одной из важнейших частей вебсайтов различной направленности. Примерами могут служить онлайн-каталог кинофильмов Imdb [1], онлайн-магазин Amazon [2], видеохостинг Youtube [3]. Одной из причин широкого