О ПОДХОДАХ К ПРОБЛЕМЕ СЖАТИЯ ДАННЫХ

Батурина. Ж.С.

Г.Бирск, ФГБОУ ВО Бирский филиал УУНиТ

Для сжатия данных было разработано множество методов. Большинство из них комбинируют различные принципы сжатия для создания полного алгоритма. Хорошие принципы в сочетании дают еще лучшие результаты. Большинство технических специалистов используют принцип энтропийного кодирования, но часто встречаются и другие – кодирование длин серий (Run-Length Encoding) и преобразование Барроуза-Уилера (Burrows-Wheeler Transform).

Одна из проблем заключается в том, что если мы оцифруем данные (звук, изображение) очень качественно, то нам понадобится очень много данных. Мы получим большие файлы, которые требуют много места для хранения, дорогие USB-накопители и большой интернет-трафик. Желательно, чтобы файл был меньшего размера. Для этого нужно использовать сжатие - различные алгоритмы, основанные на сложной математической основе и дающие на выходе данные меньшего объема.

Существует два основных типа сжатия - с потерями и без них.

Сжатие с потерями означает, что мы потеряли некоторую информацию, когда сделали это. Бесплатные алгоритмы сжатия пытаются гарантировать, что мы теряем только те данные, которые для нас не слишком важны.

Представьте, что утраченное сжатие - это краткий рассказ о произведении школьной программы: для ученика описания характера и стиля автора не так важны, для него важен основной сюжет. Краткая история сохранила только самые важные, но передала их гораздо быстрее.

Сжатие без потерь - это когда мы уменьшаем размер файла без потери качества. Для этого используются интересные математические методы и кодировки. Основная идея заключается в том, что при декодировании все данные остаются на своих местах.

Существует два основных подхода сжатия данных без потерь: алгоритм Хаффмана или алгоритм LZW. LZW используется повсеместно, но он достаточно сложен. Более прост для понимания алгоритм Хаффмана [3].

Алгоритм Хаффмана берет файл, разбивает его на фрагменты, с которыми ему удобно работать, а затем выполняет поиск по количеству повторений каждого фрагмента. Самые распространенные слова этот алгоритм обозначает коротким кодом, а самые редкие слова может оставить без изменений. Поскольку наиболее распространенные слова теперь занимают гораздо меньше места, готовый файл также становится меньше.

Однако есть и обратная сторона: иногда вам может потребоваться немедленно сохранить эту таблицу с совпадениями слов и кода в одном файле, но она может оказаться большой. В большинстве случаев алгоритм Хаффмана используется для сжатия текстовых файлов и видео без потерь.

Рассмотрим сжатие без потерь на примере звука.

В среднем одна минута несжатого звука занимает 10 мегабайт. Это довольно много: например, если у вас есть часовая запись концерта, это займет полгигабайта. С другой стороны, при такой записи запечатлены все тональности, множество высоких частот и общая красота.

Для таких ситуаций используется сжатие без потерь: оно уменьшает размер файла в 2-3 раза без искажения звука. Алгоритмы, которые сжимают звук, называются кодеками. FLAC и Apple Lossless - два популярных кодека для сжатия звука без потерь.

Оригинальный несжатый файл формата WAV займет например 23 мегабайта. Сжатый файл в формате FLAC без потерь с теми же параметрами займет 10 МБ.

Где еще применяется сжатие без потерь.

Главной задачей архивирования данных является возможность упаковать выбранные файлы таким образом, чтобы архив занимал как можно меньше места, не повреждая данные. Например, текстовая версия "Войны и мира" может занимать 4 мегабайта, в то время как архивная версия объемом 100 килобайт в 40 раз меньше.

Потеря данных может происходить и на крупных сервисах. Например Myspace потерял архивы песен за 12 лет при переносе данных с одного сервера на другой. По этому всегда должны быть резервные копии. Как правило они в сжатом виде.

Проблема сжатия данных касается и самих компьютерных программ. Существуют специальные пакеты, которые редактируют готовую программу и оптимизируют код, чтобы он занимал меньше места, но сохранял свою функциональность. Например, удаляются комментарии, минимизируются имена переменных и имена функций.

При сжатии видео и аудио часто применяют подход с потерей данных. Для обычного пользователя сервисами интернет кинотеатра качество может быть понижено. На мониторе ухудшение будет почти не заметно. Для кинотеатрального же показа конечно прибегают к меньшему сжатию видео и аудиопотока. Самый известный формат сжатия звука с потерями это MP3. Конечное качество можно выбирать.

Таким образом, в данной статье приведен краткий обзор подходов к сжатию данных с потерей и без потери информации для различных видов информации. Показано, что необходимо выбирать алгоритмы сжатия в зависимости от поставленной цели.

Литература

1. Русский язык: К успеху шаг за шагом. Способы сжатия текста без потерь: [Электронный ресурс]. URL: https://www.sites.google.com/site/russkijazykk5sagzasagom /gotovimsa - k - oge / cast -1- izlozenie /

priemy
-
szatia
-
teksta
(Дата обращения 14.03.2023)

2. Википедия. Сжатие данных: [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Сжатие_данных (Дата обращения 14.03.2023)

3. Википедия. Код Хаффмана: [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Код_Хаффмана (Дата обращения 14.03.2023)