Домашнее задание №2.

Архиватор.

Общее описание задания

Задание направленно на знакомство с различными алгоритмами сжатия данных и на анализ качества работы указанных алгоритмов. При выполнении задания можно выделить следующие основные этапы:

1. реализация базовых алгоритмов сжатия данных на основе предложенных шаблонов;
2. реализация других алгоритмов сжатия данных (бонус);
3. анализ работы реализованных алгоритмов сжатия данных;
4. написание отчета, описывающего результаты всех указанных этапов.

Задание выполнять в папке Autumn 2014 – Homework 2 в Вашей папке именной папке в Dropbox. Задание должно быть выполнено с использованием системы контроля версия Git (фактически, в этой папке Вы должны в самом начале создать Git репозиторий).

Любые вопросы по заданию присылать по электронной почте на следующий адрес: mikle.shupletsov@gmail.com.

Тема письма имеет следующий формат: [318] [Фамилия Имя] [Вопрос].

Реализация базовых алгоритмов сжатия.

В качестве базовых алгоритмов необходимо реализовать алгоритм арифметического кодирования (https://en.wikipedia.org/wiki/Arithmetic_coding) с различными моделями источника. В качестве базовых моделей источника нужно реализовать по одному из следующих классов моделей:

1. контекстная модель частичного совпадения;
2. экспертная модель;
3. композиция моделей.

Контекстные модели частичного совпадения или, иначе Марковские модели заданного порядка (https://en.wikipedia.org/wiki/Prediction_by_partial_matching). Предполагается реализация необходимо реализовать контекстную модель заданного порядка, например, PPM(2) или PPM(3). При этом стоит учитывать, что в распоряжении студентов есть шаблон заготовка, в которой присутствует базовая реализация PPM(0) и PPM(1).

При реализации экспертной модели студент выступает в роли эксперта, который обладает специальным знанием о структуре входной последовательности. Например, если входная последовательность представляет текст на русском языке, то это может быть информация о структуре слов (например, какие сочетания букв могут встретиться в языке, а какие нет) или часто встречающихся фрагментов (например, приставки и суффиксы). При этом единственное ограничение, которое накладывается на формулировку модели, заключается в том, что модель должна выдавать распределение вероятностей символов или условной вероятности символа, при реализации заданного контекста.

В качестве композиции моделей требуется реализовать один из следующих двух типов композиции:

1. алгебраическая;
2. логическая.

Алгебраическая композиция представляет такую композицию моделей, при которой все базовые модели выдают свое распределение вероятностей, а потом в рамках корректирующей операции производится свертка указанных распределений в одно. Логическая композиция может быть реализована при помощи техники уходов (escapes).

Реализация на C++ с использованием заготовки

Требуется реализовать алгоритмы на языке C++. При этом реализация на языке C++ должна быть сделана с использованием шаблона, размещенного в Вашей папке. Указанный шаблон содержит реализацию алгоритма арифметического кодирования и базовых PPM моделей ранга -1, 0 и 1.

Заготовка является Вашей отправной точкой в выполнении задания. Представьте, что Вам был передан частично завершенный проект, который Вам требуется закончить. Поэтому код, написанный Вами, должен быть согласован с изначальной заготовкой. Изменения, вносимые в саму заготовку, должны иметь разумные причины и соответствующим образом задокументированы.

Реализация других алгоритмов сжатия (бонус)

Дополнительно, можно реализовать другие алгоритмы сжатия данных или адаптивные модели источников для алгоритма арифметического кодирования. Соответствующий алгоритм или модель должны быть подробно описаны в тексте отчета, и должно быть проведено тестирование качества сжатия полученной при этом реализации.

Замеры скорости работы и анализ качества сжатия реализованных алгоритмов сжатия

Тестирование полученных реализаций алгоритмов сжатия данных требуется протестировать на следующих типах файлов:

1 текст на русском языке в кодировке cp1251;
2 текст на русском языке в кодировке UTF-8;
3 текст на английском языке (в кодировке cp1251 или UTF-8, в данном случае они неразличимы);
4 данные в формате XML.

Студентам предлагается самостоятельно подобрать файлы для тестирования (например, это могут быть фрагменты текстов ваших любимых книг). При этом тестовая выборка формируется по следующим правилам. Сначала генерируется по одному файлу каждого из четырех представленных типов размером 1Мб. Далее, каждый файл разбивается на два файла по 512Кб (пополам). При этом каждый фрагмент дополнительно инвертируется. В итоге получается 16 файлов размером 512Кб (половина файлов содержит фрагменты исходных текстов, другая половина содержит указанные фрагменты в инвертированным виде). Тестовая выборка состоит из файлов размером 1Мб, полученной из указанных 16 файлов объединением(конкатенацией) произвольной пары файлов.

Независимо от тестирования, проводимого студентами, реализации алгоритмов будут тестироваться на скрытой выборке преподавателями. На основе результатов скрытого тестирования будет проведен конкурс. Стоит отметить, что результаты конкурса не влияют на оценку задания.

Требования к отчету

Отчет по заданию должен содержать следующие основные разделы:

1. Введение.
2. Постановка задачи.
3. Описание алгоритма сжатия данных и адаптивных моделей источника.
4. Результаты тестирования алгоритмов и анализ качества сжатия данных.
5. Использование системы контроля версий.
6. Заключение.
7. Список литературы.

Раздел «Введение» кратко описывает суть практического задания и его мотивацию.

Раздел «Постановка задачи» описывает формальную постановку задачи с учетом выбранных подходов к построению алгоритма сжатия данных и адаптивных моделей источника.

Раздел «Описание алгоритма сжатия данных и адаптивных моделей источника» содержит подробное описание выбранного алгоритма (или алгоритмов) сжатия данных и формальное описание используемых адаптивных моделей источника с указанием основных параметров и стратегии обновления модели.

Раздел «Результаты тестирования алгоритмов и анализ качества сжатия данных» содержит описание того, как тестировалась программа. Должна быть представлена информация об используемой тестовой выборке. Должен быть проведен анализ качества сжатия на тестовой выборке с построением соответствующих графиков, иллюстрирующих качество сжатия данных (замеры производить каждые 256 символов входного файла).

Раздел «Использование системы контроля версий» содержит описание того, как система контроля версий использовалась при выполнении задания. Кроме того, должна быть приведена статистика по числу коммитов, ветвей и других показателей, которые можно рассчитать при использовании системы контроля версий.

Раздел «Заключение» кратко описывает результаты практической работы, а также выводы, которые можно сделать из результатов тестирования.

Раздел «Список литературы» содержит ссылки на статьи и электронные ресурсы, если таковые были упомянуты в тексте отчета.

Критерии оценки

Результаты выполнения домашнего задания оцениваются по следующим основным критериям:

1. корректность и качество реализации моделей источников для сжатия данных;
2. использование системы контроля версий;
3. качество оформления кода (styleguide);
4. тестирование программы и анализ производительности;
5. документирование кода;
6. структура и содержание отчета.