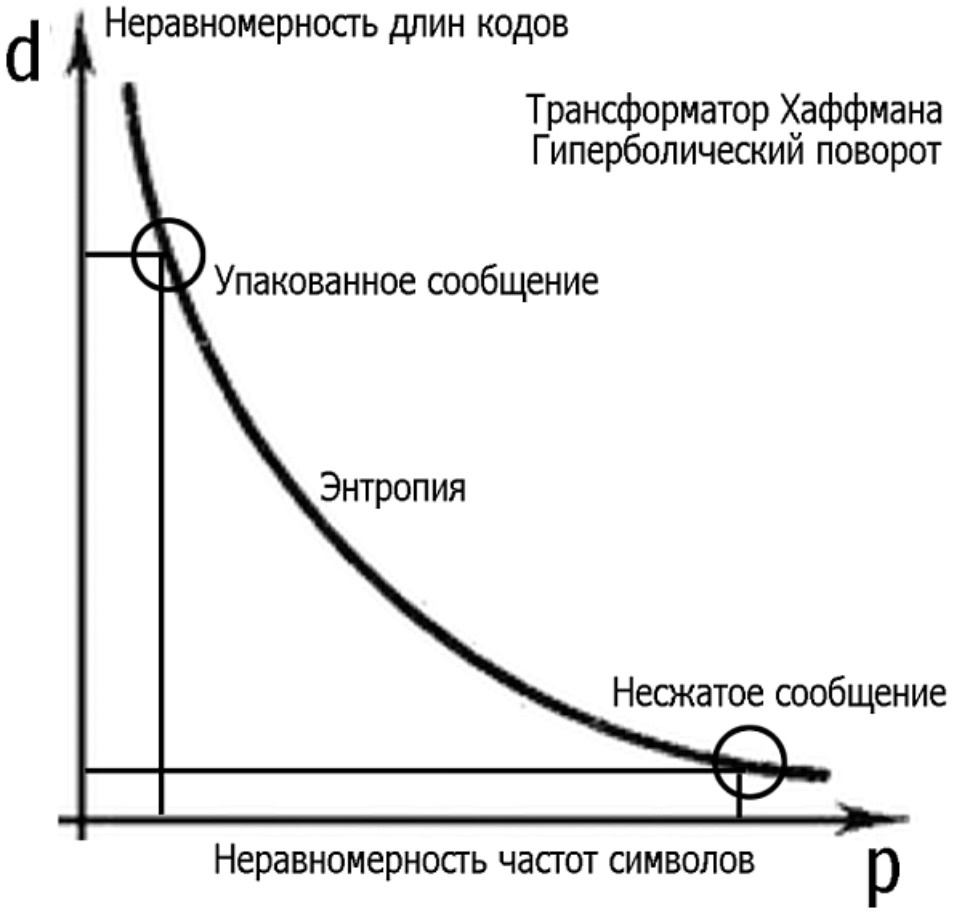
Трансформатор Хаффмана - теоретическая часть



Однажды Ходжу Насреддина спросили: - Как строятся самые высокие минареты? - Очень просто, - ответил тот. - Роют глубокий колодец, а потом выворачивают его наизнанку.

Алгоритм компрессии данных Давида Хаффмана ("Трансформатор Хаффмана") описан в сотнях (если не тысячах) статей, но я не знаю ни одной, где это было бы сделано правильно :-)

Во-первых, алгоритм Хаффмана связывают исключительно с текстом. Между тем, текст - один из вариантов представления фрактала (для простоты не будем различать истинные - бесконечные фракталы и конечные предфракталы). То, что текст - фрактальный объект следует, например, из его эквивалентного представления другим классическим фрактальным объектом - деревом.

Трансформация Хаффмана применима к произвольному фрактальному объекту. Такой объект может быть отображен на другой (в том числе, двойственный) фрактальный объект с иным правилом разбиения. Выбор минимального набора элементов (например, минимального числа разновесов при взвешивании) отвечает

оптимальной стратегии разбиения и эквивалентен алгоритму Хаффмана.

Во-вторых, при описании алгоритма Хаффмана умалчивается, что вторая фаза алгоритма является инверсией построенного на первом шаге дерева максимальной высоты.

Это затемняет тот факт, что с инверсией дерева связано понятие инварианта трансформации - в терминологии Клода Шеннона "количество информации" в сообщении.

Два экстремальных (двойственных) варианта полностью сбалансированного дерева - это симметричное дерево минимальной высоты, обычно, называемое просто "сбалансированным деревом", высота которого есть (двоичный) логарифм от числа терминальных узлов ("дерево Хартли") и дерево максимальной высоты.

Инверсия Хаффмана заключается в сопоставлении каждому терминальному узлу построенного в первой фазе алгоритма дерева максимальной высоты инцидентной к нему ветви. При этом каждому терминальному узлу ставится в соответствие код этой ветви таким образом, что узел с наибольшим весом (самый частый символ) получает наиболее короткий код. В результате, код этого узла имеет наименьшую долю в полном кодовом пространстве и, наоборот, код с наименьшим весом (самый редкий символ) получает наибольшую долю. В результате такого выравнивания происходит симметрирование кодового дерева и, в пределе, инверсии дерева соответствует двойственное ему дерево Хартли.

Это ключевой момент алгоритма: спектру кратностей символов ставится в соответствие спектр битовых размеров кодов. Чем лучше согласованы эти спектры (согласование источника сигнала и канала связи), тем выше компрессия. В этом смысле, устройство компрессии данных полностью эквивалентно обычному согласующему трансформатору.

На практике, спектр битовых размеров кодов, обычно, жестко задан (например, размером машинного слова), а на спектр кратностей символов можно повлиять выбором подходящей модели парсинга текста.

Нетрудно видеть, что при минимальном размере кода в один бит, предельная степень сжатия не может превышать битового размера символа исходного алфавита и линейно растет с ростом этого размера. Иными словами, паковать выгодно не отдельные символы, а крупные блоки (слова, фразы).

Энтропия Шеннона описывает "насыпную" (фрактальную) плотность текста, который можно "утрамбовать" до энтропийного предела. В результате "сжатия" сообщения (термин неверный, но общепринятый) его размер уменьшается, при этом энтропия на символ текста растет, и, в результате, "количество информации" остается неизменным. Таким образом, на плоскости параметров (размер сообщения, энтропия на символ) сжатие текста отвечает гиперболическому повороту (лоренц-сжатие) и преобразования текста могут быть описаны в терминах, используемых в теории относительности.