Информатика

Учебный год 2015/2016

Кафедра ВТ Университета ИТМО

Соснин В.В., Балакшин П.В.

Лекция 4. Сжатие данных (код Хаффмана)

Дайджест аннотаций студентов

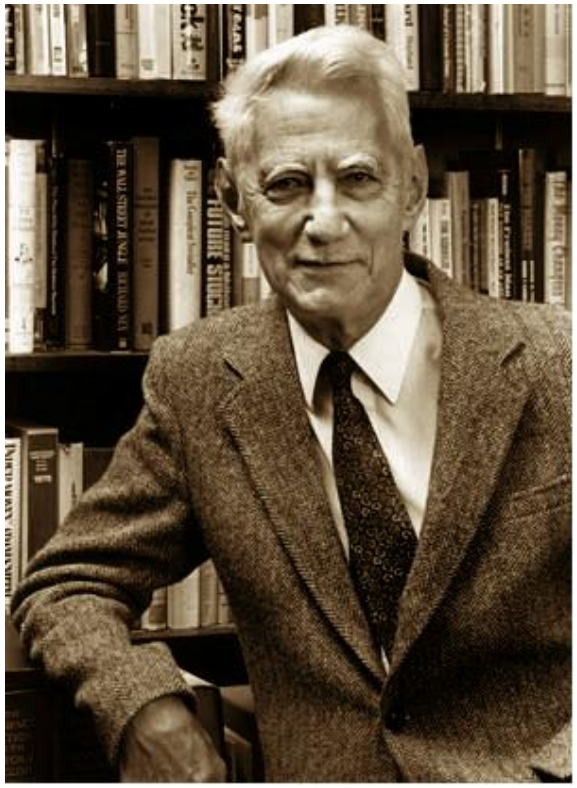
1. Крыса управляет истребителем.
2. Самонаводящаяся пуля.
3. Тепло тела → батарейка.
4. SD-карта: 512 ГБ, 90 МБ/с, 30 000 руб.
5. Технология 4D-печати.
6. Робот для защиты гос. границы.
7. Восьмиядерный процессор для ПК.
8. Замена кремниевым транзисторам (солевая грелка).
9. Голографические 3D-дисплеи.
10. WD10TB, GTX980.

Дайджест аннотаций студентов

Критерий оценивания аннотаций

1. Энтузиазм/фанатизм учитывается.
2. Умение видеть ± — это плюс.
3. Не надо новости IT-экономики.
4. Статьи на английском — very well!
5. «Автор Анонимус» = «надпись на заборе».
6. Видеолекции приветствуются!

Определение



Клод Шеннон (1916-2001)

Сжатие данных –
это процесс, обеспечивающий уменьшение объёма данных путем сокращения их избыточности.

К. Шеннон

Сжатие данных – это частный случай *кодирования* данных.

Кодирование

Кодирование – процесс преобразования символов алфавита X в символы алфавита Y. **Декодирование** – обратный процесс. При этом наименьшая единица данных, рассматриваемая как единое целое при кодировании/декодировании – это символ.

Кодовое слово – последовательность символов из алфавита Y, однозначно обозначающая конкретный символ алфавита X.

Средняя длина кодового слова – это величина, которая вычисляется как взвешенная вероятностями сумма длин всех кодовых слов.

Если все кодовые слова имеют одинаковую длину, то код называется ***равномерным*** *(фиксированной длины). Е*сли встречаются слова разной длины, то – ***неравномерным*** *(переменной длины)*.

Характеристики кодирования

$$\textbf{Коэффициент сжатия} = \frac{\text{Размер } \textbf{вх}\text{одного потока}}{\text{Размер } \textbf{вых}\text{одного потока}}$$

$$\textbf{Отношение сжатия} = \frac{\text{Размер } \textbf{вых}\text{одного потока}}{\text{Размер } \textbf{вх}\text{одного потока}}$$

Виды сжатия данных

1. Сжатие без потерь *(полностью обратимое):* сжатые данные после декодирования (распаковки) не отличаются от исходных.

2. Сжатие с потерями (*частично обратимое*) сжатые данные после декодирования (распаковки) отличаются от исходных, т.к. при сжатии часть исходных данных была отброшена для увеличения коэффициента сжатия.

Примеры и краткая характеристика методов сжатия

1. Метод кодирования длины серий.
2. Метод кодирования по словарю.
3. Энтропийное кодирование.
4. …

Префиксный код – это код, в котором никакое кодовое слово не является префиксом любого другого кодового слова. Эти коды имеют переменную длину.

Оптимальный префиксный код – это префиксный код, имеющий минимальную среднюю длину.

Алгоритм Шеннона-Фано

Дана последовательность символов:
AAABCCCCDEEEFG

p(A) = 3/14, p(B) = 1/14, p(C) = 4/14, p(D) = 1/14
p(E) = 3/14, p(F) = 1/14, p(G) = 1/14

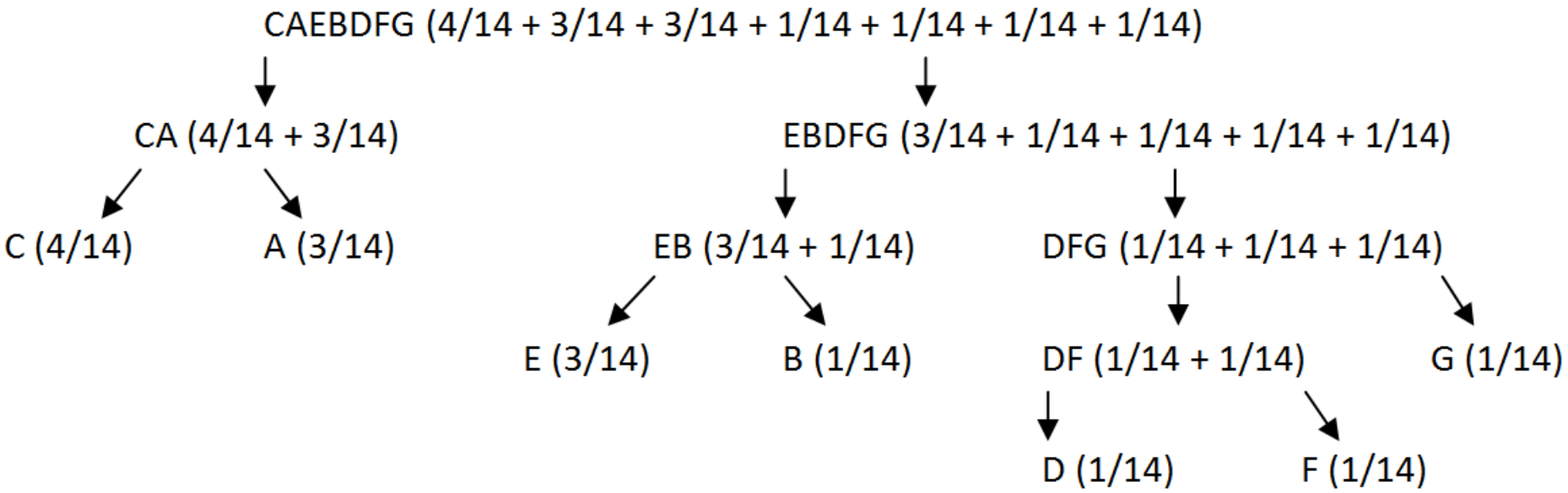
Отсортируем таблицу в порядке убывания вероятности символов:

Роберт Фано (род. 1917)

Символ	Вероятность
C	4/14
A	3/14
E	3/14
B	1/14
D	1/14
F	1/14
G	1/14

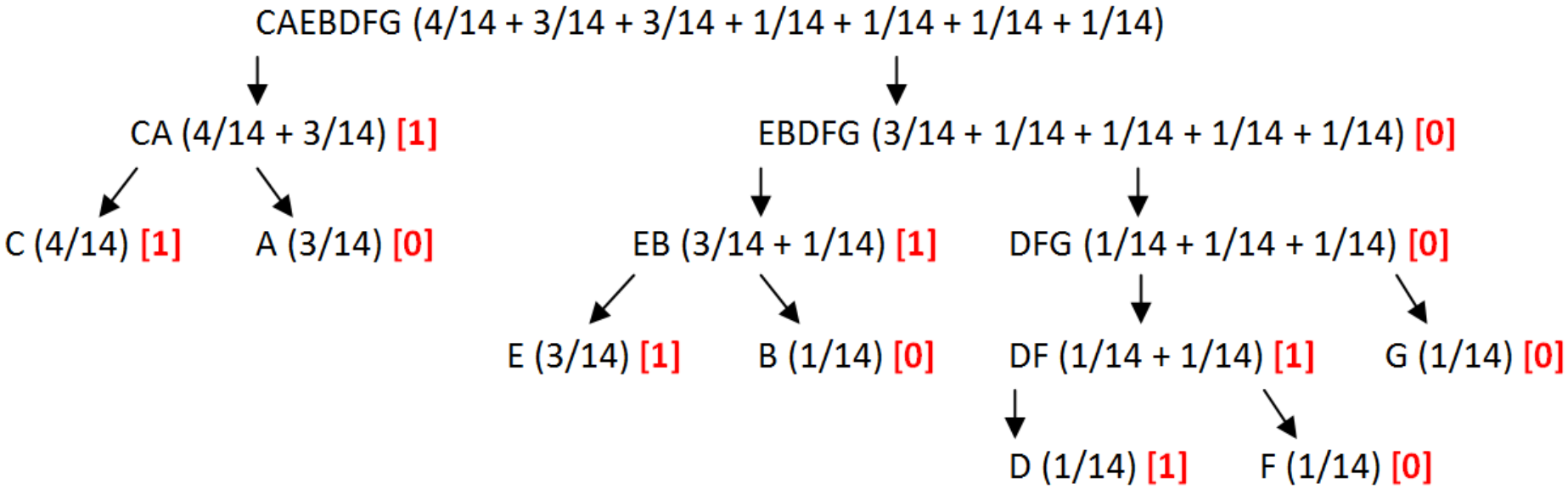
Алгоритм Шеннона-Фано (2)

Построим кодовое дерево от корня к листьям (сверху вниз)



Алгоритм Шеннона-Фано (3)

Левому символу (с большей вероятностью) присвоим значение 1, правому – 0.



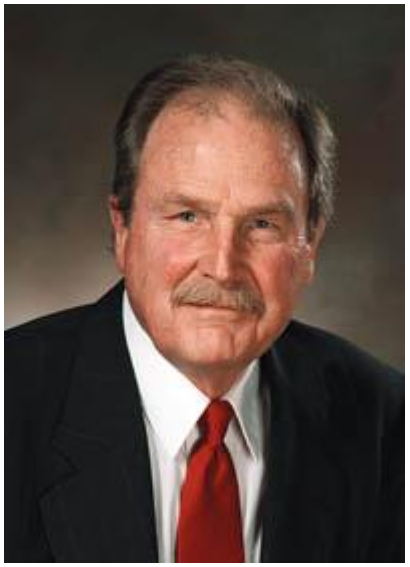
Алгоритм Шеннона-Фано (4)

Получим следующую таблицу для кодировки:

Символ	Вероятность	Код
C	4/14	11
A	3/14	10
E	3/14	011
B	1/14	010
D	1/14	0011
F	1/14	0010
G	1/14	000

Исходная последовательность AAABCCCCDEEEFG кодируется следующей :
10.10.10.010.11.11.11.11.0011.011.011.011.0010.000 – 37 бит

Алгоритм (код) Хаффмана



Дэвид Хаффман (1925-1999) – аспирант Фано

Код Хаффмана

Сжатие данных по Хаффману применяется при сжатии фото- и видеоизображений (JPEG, стандарты сжатия MPEG), в архиваторах (PKZIP, LZH), в протоколах передачи данных MNP5 и MNP7.

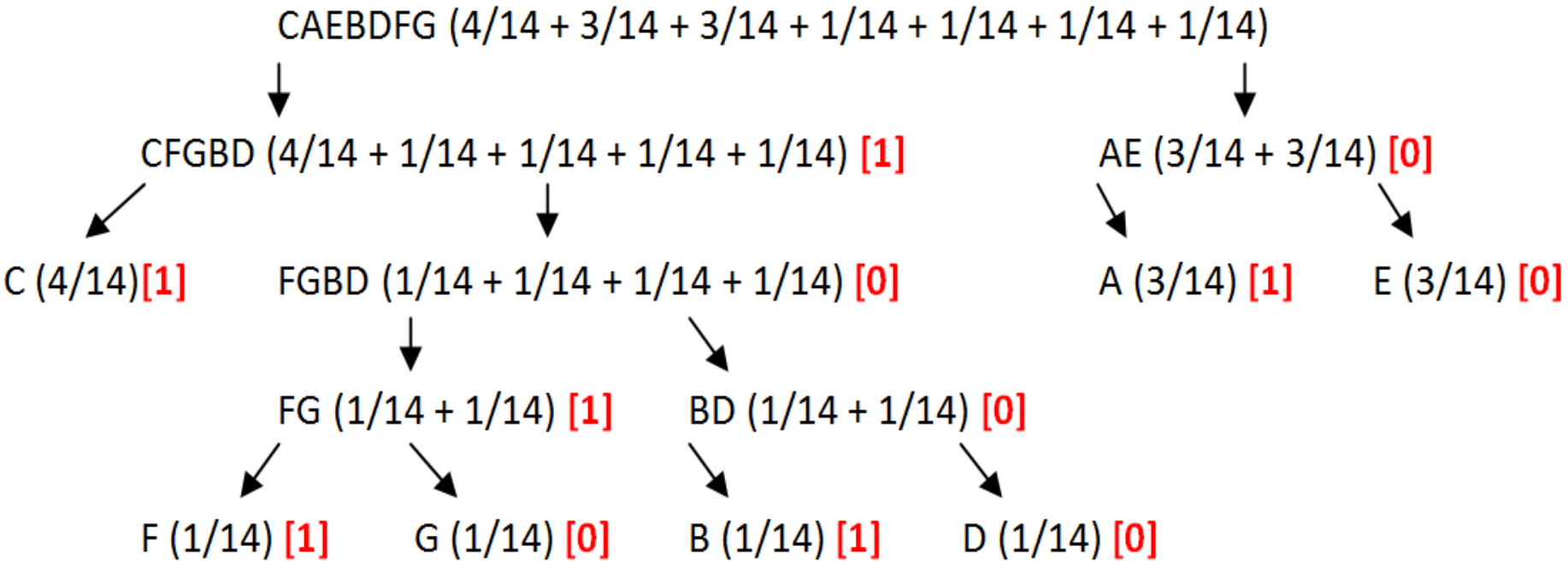
Построение дерева по методу Хаффмана

1. Символы входного алфавита образуют список свободных узлов. Каждый узел имеет вес, равный вероятности появления символа в сжимаемом тексте.
2. Выбираются два свободных узла дерева с наименьшими весами.
3. Создается их родитель с весом, равным их суммарному весу. После этого шага родитель рассматривается как свободный узел.
4. Родитель добавляется в список свободных узлов, а двое его детей удаляются из этого списка.
5. Одной дуге, выходящей из родителя, ставится в соответствие бит 1, другой – бит 0 (*для определённости можно считать, что к узлу с бóльшим весом соответствует 1, с меньшим – 0).*
6. Повторяем шаги 2-6, пока в списке свободных узлов не останется только один свободный узел. Он и будет считаться корнем дерева.

Пример построения дерева по методу Хаффмана

Сжимаемое сообщение – AAABCCCCDEEEFG (то же, что при использовании алгоритма Шеннона-Фано). Соответственно: p(A) = 3/14, p(B) = 1/14, p(C) = 4/14, p(D) = 1/14, p(E) = 3/14, p(F) = 1/14, p(G) = 1/14.

Построим кодовое дерево (снизу вверх)



Построение кода Хаффмана по дереву

Получим следующую таблицу для кодировки:

Символ	Вероятность	Код
C	4/14	11
A	3/14	01
E	3/14	00
B	1/14	1001
D	1/14	1000
F	1/14	1011
G	1/14	1010

Исходная последовательность AAABCCCCDEEEFG кодируется следующей:
01.01.01.1001.11.11.11.11.1000.00.00.00.1011.1010 – 36 бит (это лучше, чем в коде Шеннона-Фано)

Построение кода Хаффмана без дерева

Выберем 2 элемента с минимальной вероятностью. Формируем новый узел с вероятностью, равной сумме предыдущих 2 элементов. Полученная сумма становится новым элементом таблицы, занимающим соответствующее место в списке убывающих по величине вероятностей. Процедура продолжается до тех пор, пока в таблице не останутся всего два элемента.

Символ	**Вероят-ность, p**	**Символ**	**p1**	**Символ**	**p2**	**Символ**	**p3**	**Символ**	**p4**	**Символ**	**p5**
C	4/14	C	4/14	C	4/14	C	4/14	AE	6/14	CFGBD	8/14
A	3/14	A	3/14	A	3/14	FGBD	4/14	C	4/14	AE	6/14
E	3/14	E	3/14	E	3/14	A	3/14	FGBD	4/14		
B	1/14	FG	2/14	FG	2/14	E	3/14				
D	1/14	B	1/14	BD	2/14						
F	1/14	D	1/14								
G	1/14										

Построение кода Хаффмана без дерева

Символ	Вероят-ность, p	Символ	p1	Символ	p2	Символ	p3	Символ	p4	Символ	p5
C	4/14	C	4/14	C	4/14	C	4/14	AE	6/14	CFGBD [1]	8/14
A	3/14	A	3/14	A	3/14	FGBD	4/14	C [1]	4/14	AE [0]	6/14
E	3/14	E	3/14	E	3/14	A [1]	3/14	FGBD [0]	4/14		
B	1/14	FG	2/14	FG [1]	2/14	E [0]	3/14				
D	1/14	B [1]	1/14	BD [0]	2/14						
F [1]	1/14	D [0]	1/14								
G [0]	1/14										

Результирующий код Хаффмана будет иметь такой же коэффициент сжатия, как и при использовании дерева! Алгоритм Хаффмана одинаково работает и с таблицей, и с деревом!