Теория информации

Кодирование без потери информации
Часть I. Теорема Крафта–Макмиллана

Михаил Владимирович Волков
Уральский федеральный университет

Вводные замечания

Количество материала в этой части примерно соответствует двум офлайновым лекциям.

Вводные замечания

Количество материала в этой части примерно соответствует двум офлайновым лекциям.

Конечно, прослушать (а тем более проработать) этот материал за один присест нереально. Делайте паузы.

Вводные замечания

Количество материала в этой части примерно соответствует двум офлайновым лекциям.
Конечно, прослушать (а тем более проработать) этот материал за один присест нереально. Делайте паузы.

Я не стал нумеровать слайды, но снабдил каждый слайд уникальным заголовком.

Вводные замечания

Количество материала в этой части примерно соответствует двум офлайновым лекциям.
Конечно, прослушать (а тем более проработать) этот материал за один присест нереально. Делайте паузы.

Я не стал нумеровать слайды, но снабдил каждый слайд уникальным заголовком.
Когда/если появятся вопросы и/или замечания в связи с каким-то местом в лекциях, используйте заголовок слайда для локализации этого места.

Вводные замечания

Количество материала в этой части примерно соответствует двум офлайновым лекциям.
Конечно, прослушать (а тем более проработать) этот материал за один присест нереально. Делайте паузы.

Я не стал нумеровать слайды, но снабдил каждый слайд уникальным заголовком.
Когда/если появятся вопросы и/или замечания в связи с каким-то местом в лекциях, используйте заголовок слайда для локализации этого места.
Например: «Почему на слайде ‘Сигнальные флаги’ одной и той же цифре 2 в коде 220 слова ‘do’ отвечают два разных флага? Это опечатка?»

Вводные замечания

Количество материала в этой части примерно соответствует двум офлайновым лекциям.
Конечно, прослушать (а тем более проработать) этот материал за один присест нереально. Делайте паузы.

Я не стал нумеровать слайды, но снабдил каждый слайд уникальным заголовком.
Когда/если появятся вопросы и/или замечания в связи с каким-то местом в лекциях, используйте заголовок слайда для локализации этого места.
Например: «Почему на слайде ‘Сигнальные флаги’ одной и той же цифре 2 в коде 220 слова ‘do’ отвечают два разных флага? Это опечатка?»

В тексте есть изредка встречаются вопросы/упражнения.
При первом чтении/прослушивании можно их пропускать, но полезно к ним вернуться позже и попробовать найти ответ/решение!

Для чего нужно понятие энтропии?

В предыдущих лекциях мы дали аксиоматическое определение энтропии, вывели из аксиом формулу для энтропии

$$H(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log p_i,$$

$$\text{где } \; 0 \leqslant p_1, \ldots, p_n \leqslant 1, \; \sum_{i=1}^{n} p_i = 1,$$

и познакомились с основными свойствами функции H.

Для чего нужно понятие энтропии?

В предыдущих лекциях мы дали аксиоматическое определение энтропии, вывели из аксиом формулу для энтропии

$$H(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log p_i,$$

$$\text{где} \quad 0 \leqslant p_1, \ldots, p_n \leqslant 1, \ \sum_{i=1}^{n} p_i = 1,$$

и познакомились с основными свойствами функции H.

Осталось понять, для чего это понятие может быть полезно.

Для чего нужно понятие энтропии?

В предыдущих лекциях мы дали аксиоматическое определение энтропии, вывели из аксиом формулу для энтропии

$$H(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log p_i,$$

$$\text{где} \quad 0 \leqslant p_1, \ldots, p_n \leqslant 1, \ \sum_{i=1}^{n} p_i = 1,$$

и познакомились с основными свойствами функции H.

Осталось понять, для чего это понятие может быть полезно.

В этих слайдах рассказывается о применении энтропии в задаче сжатия информации без потерь.

Трафальгарская битва

Произошла 21 октября 1805 года у мыса Трафальгар на атлантическом побережье Испании около города Кадис.

Трафальгарская битва

Произошла 21 октября 1805 года у мыса Трафальгар на атлантическом побережье Испании около города Кадис.

Английский флот под командованием адмирала Горацио Нельсона разгромил франко-испанскую эскадру и тем самым похоронил планы Наполеона высадить десант в Англии.

Трафальгарская битва

Произошла 21 октября 1805 года у мыса Трафальгар на атлантическом побережье Испании около города Кадис.

Английский флот под командованием адмирала Горацио Нельсона разгромил франко-испанскую эскадру и тем самым похоронил планы Наполеона высадить десант в Англии.

Адмирал Нельсон был убит в этой битве.

Трафальгарская битва

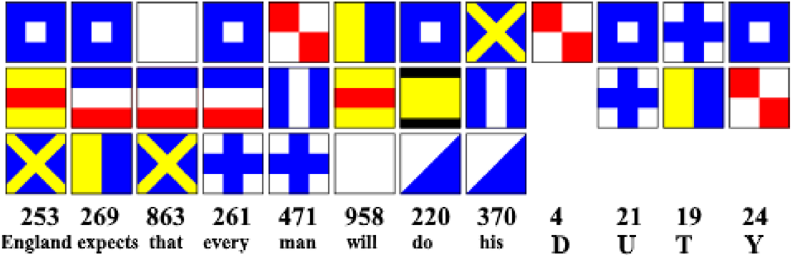
Произошла 21 октября 1805 года у мыса Трафальгар на атлантическом побережье Испании около города Кадис.

Английский флот под командованием адмирала Горацио Нельсона разгромил франко-испанскую эскадру и тем самым похоронил планы Наполеона высадить десант в Англии.

Адмирал Нельсон был убит в этой битве.

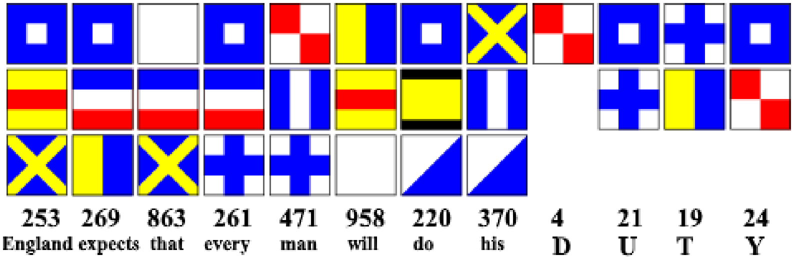
Под началом Нельсона было 27 линейных кораблей, 4 фрегата и 2 шлюпа. Как он командовал своим флотом?

Сигнальные флаги



Команды передавались с флагманского линкора «Victory» с помощью сигнальных флагов (их поднимали на канате, прикрепленном к мачте).

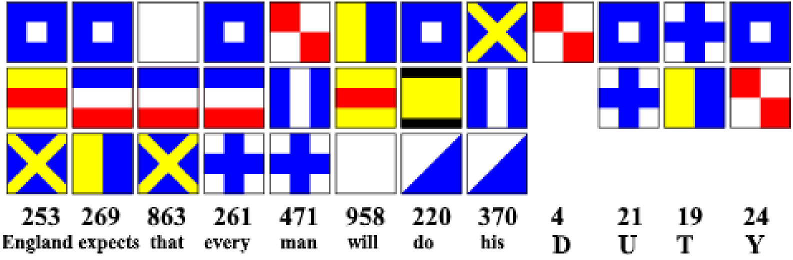
Сигнальные флаги



Команды передавались с флагманского линкора «Victory» с помощью сигнальных флагов (их поднимали на канате, прикрепленном к мачте).

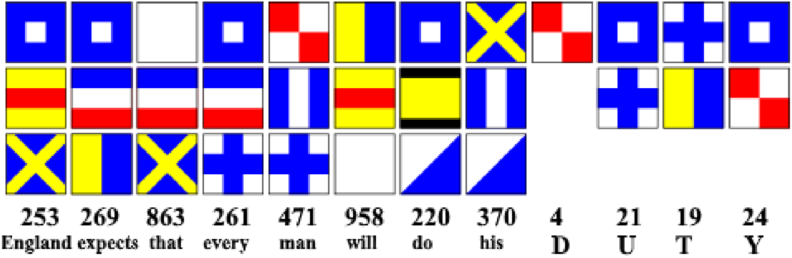
На картинке показан знаменитый приказ Нельсона
ENGLAND EXPECTS THAT EVERY MAN WILL DO HIS DUTY

Сжатие для ускорения передачи



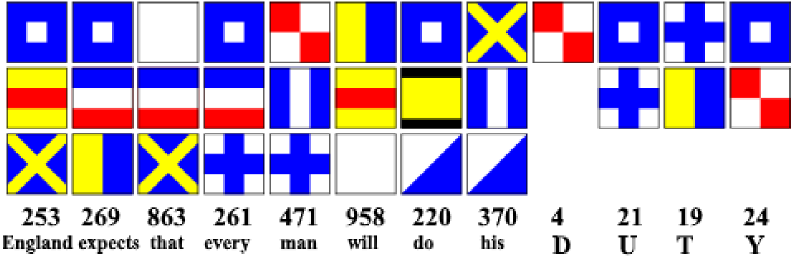
Отчет офицера связи:

Сжатие для ускорения передачи



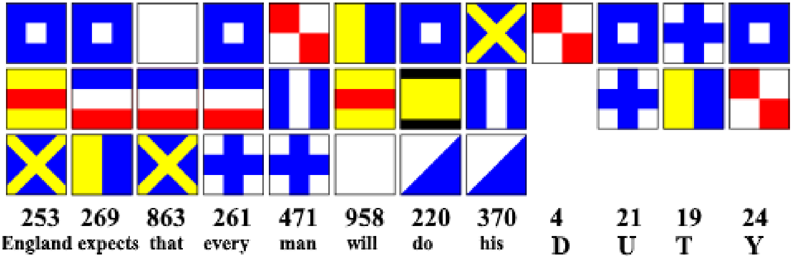
Отчет офицера связи: His Lordship . . . said, 'Mr. Pasco, I wish to say to the fleet, ENGLAND CONFIDES THAT EVERY MAN WILL DO HIS DUTY'

Сжатие для ускорения передачи



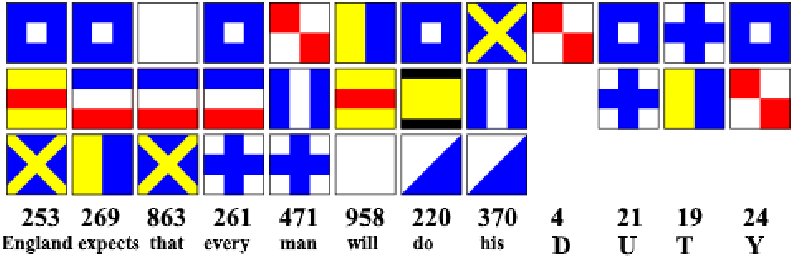
Отчет офицера связи: His Lordship ... said, 'Mr. Pasco, I wish to say to the fleet, ENGLAND CONFIDES THAT EVERY MAN WILL DO HIS DUTY' and he added 'You must be quick, for I have one more to make which is for close action.'

Сжатие для ускорения передачи



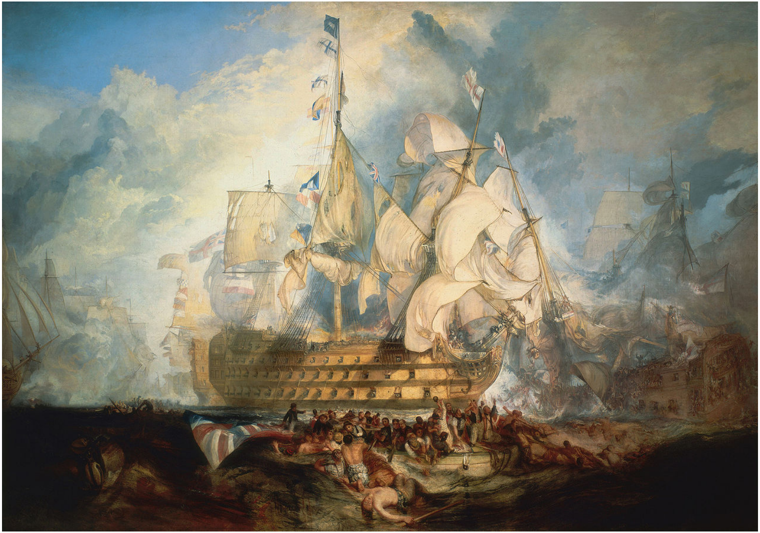
Отчет офицера связи: His Lordship . . . said, 'Mr. Pasco, I wish to say to the fleet, ENGLAND CONFIDES THAT EVERY MAN WILL DO HIS DUTY' and he added 'You must be quick, for I have one more to make which is for close action.' I replied, 'If your Lordship will permit me to substitute the CONFIDES for EXPECTS the signal will soon be completed, because the word EXPECTS is in the vocabulary, and CONFIDES must be spelt.'

Сжатие для ускорения передачи



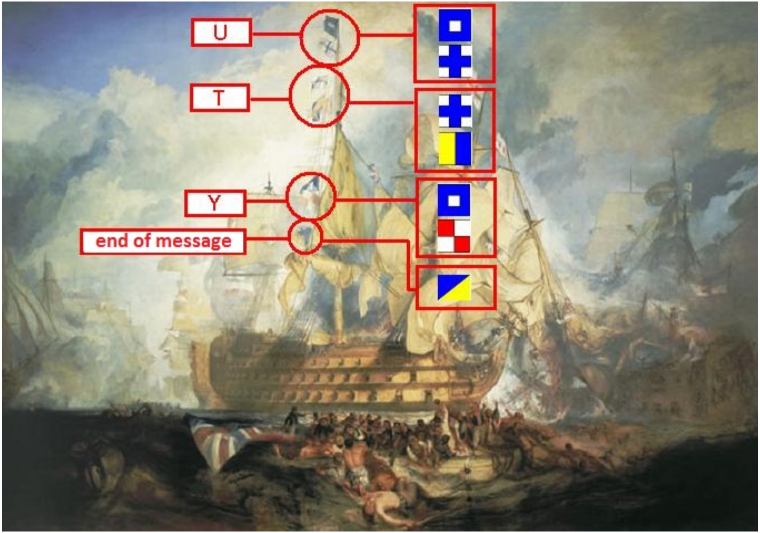
Отчет офицера связи: His Lordship . . . said, ’Mr. Pasco, I wish to say to the fleet, ENGLAND CONFIDES THAT EVERY MAN WILL DO HIS DUTY’ and he added ’You must be quick, for I have one more to make which is for close action.’ I replied, ’If your Lordship will permit me to substitute the CONFIDES for EXPECTS the signal will soon be completed, because the word EXPECTS is in the vocabulary, and CONFIDES must be spelt.’ His Lordship replied, in haste, and with seeming satisfaction, ’That will do, Pasco, make it directly.’

Сжатие для ускорения передачи, продолжение



Глядя на картину Тернера «Трафальгарская битва», легко понять, сколь существенно ускорение, предложенное Паско!

Сжатие для ускорения передачи, продолжение



Глядя на картину Тернера «Трафальгарская битва», легко понять, сколь существенно ускорение, предложенное Паско!

Основная идея

Для сжатия нужно кодировать часто встречающиеся знаки и/или знакосочетания короткими последовательностями, а редко встречающиеся – длинными.

Основная идея

Для сжатия нужно кодировать часто встречающиеся знаки и/или знакосочетания короткими последовательностями, а редко встречающиеся – длинными.

Эта идея была применена еще в классическом коде Морзе

Основная идея

Для сжатия нужно кодировать часто встречающиеся знаки и/или знакосочетания короткими последовательностями, а редко встречающиеся – длинными.

Эта идея была применена еще в классическом коде Морзе (Морзе пересчитал литеры в наборной кассе типографии):

A	· —	U	· · —
B	— · · ·	V	· · · —
C	— · — ·	W	· — —
D	— · ·	X	— · · —
E	·	Y	— · — —
F	· · — ·	Z	— — · ·
G	— — ·		
H	· · · ·		
I	· ·		
J	· — — —		
K	— · —	1	· — — — —
L	· — · ·	2	· · — — —
M	— —	3	· · · — —
N	— ·	4	· · · · —
O	— — —	5	· · · · ·
P	· — — ·	6	— · · · ·
Q	— — · —	7	— — · · ·
R	· — ·	8	— — — · ·
S	· · ·	9	— — — — ·
T	—	0	— — — — —

Основные вопросы

Есть ли какая-то предельная степень сжатия без потери информации, которой можно добиться?

Основные вопросы

Есть ли какая-то предельная степень сжатия без потери информации, которой можно добиться?

Если есть, то как ее вычислить?

Основные вопросы

Есть ли какая-то предельная степень сжатия без потери информации, которой можно добиться?

Если есть, то как ее вычислить?

И как построить то оптимальное кодирование, при котором достигается предельно возможное сжатие?

Основные вопросы

Есть ли какая-то предельная степень сжатия без потери информации, которой можно добиться?

Если есть, то как ее вычислить?

И как построить то оптимальное кодирование, при котором достигается предельно возможное сжатие?

Ответы на эти вопросы были найдены на рубеже 1940/50-х гг. Интересно, что ключевые результаты были получены студентами-дипломниками.

Основные вопросы

Есть ли какая-то предельная степень сжатия без потери информации, которой можно добиться?

Если есть, то как ее вычислить?

И как построить то оптимальное кодирование, при котором достигается предельно возможное сжатие?

Ответы на эти вопросы были найдены на рубеже 1940/50-х гг. Интересно, что ключевые результаты были получены студентами-дипломниками.

«Лучшие идеи приходят из университетов, из мест, полных молодостью, знаниями и свободой» – Эрик Шмидт, генеральный директор компании Google в 2001–2011 гг.

Кодирование

Сначала уточним, что такое кодирование.

Кодирование

Сначала уточним, что такое кодирование.

Пусть A и X – два конечных алфавита (алфавит исходных знаков и алфавит кодовых букв).

Кодирование

Сначала уточним, что такое кодирование.

Пусть A и X – два конечных алфавита (алфавит исходных знаков и алфавит кодовых букв).
Через A^+ и X^+ обозначаем множества всех непустых конечных последовательностей знаков из A и соответственно букв из X.

Кодирование

Сначала уточним, что такое кодирование.

Пусть A и X – два конечных алфавита (алфавит исходных знаков и алфавит кодовых букв).
Через A^+ и X^+ обозначаем множества всех непустых конечных последовательностей знаков из A и соответственно букв из X.
Удобно последовательности из A^+ называть текстами,
а последовательности из X^+ – словами.

Кодирование

Сначала уточним, что такое кодирование.

Пусть A и X – два конечных алфавита (алфавит исходных знаков и алфавит кодовых букв).
Через A^+ и X^+ обозначаем множества всех непустых конечных последовательностей знаков из A и соответственно букв из X.
Удобно последовательности из A^+ называть текстами,
а последовательности из X^+ – словами.

Кодирование A с помощью X – это такое отображение $\varphi\colon A \to X^+$, что его продолжение $\bar{\varphi}$ на A^+ по правилу:

$$\bar{\varphi}(a_1 a_2 \cdots a_\ell) := \varphi(a_1)\varphi(a_2)\cdots\varphi(a_\ell)$$

взаимно однозначно.

Кодирование

Сначала уточним, что такое кодирование.

Пусть A и X – два конечных алфавита (алфавит исходных знаков и алфавит кодовых букв).
Через A^+ и X^+ обозначаем множества всех непустых конечных последовательностей знаков из A и соответственно букв из X.
Удобно последовательности из A^+ называть текстами,
а последовательности из X^+ – словами.

Кодирование A с помощью X – это такое отображение $\varphi\colon A \to X^+$, что его продолжение $\bar{\varphi}$ на A^+ по правилу:

$$\bar{\varphi}(a_1 a_2 \cdots a_\ell) := \varphi(a_1)\varphi(a_2)\cdots\varphi(a_\ell)$$

взаимно однозначно. Иначе говоря, по каждому слову $y \in \bar{\varphi}(A^+)$ однозначно восстанавливается тот текст $w \in A^+$, для которого $\bar{\varphi}(w) = y$.

Кодирование

Сначала уточним, что такое кодирование.

Пусть A и X – два конечных алфавита (алфавит исходных знаков и алфавит кодовых букв).
Через A^+ и X^+ обозначаем множества всех непустых конечных последовательностей знаков из A и соответственно букв из X.
Удобно последовательности из A^+ называть текстами, а последовательности из X^+ – словами.

Кодирование A с помощью X – это такое отображение $\varphi\colon A \to X^+$, что его продолжение $\bar{\varphi}$ на A^+ по правилу:

$$\bar{\varphi}(a_1 a_2 \cdots a_\ell) := \varphi(a_1)\varphi(a_2)\cdots\varphi(a_\ell)$$

взаимно однозначно. Иначе говоря, по каждому слову $y \in \bar{\varphi}(A^+)$ однозначно восстанавливается тот текст $w \in A^+$, для которого $\bar{\varphi}(w) = y$. Множество $\varphi(A)$ называют кодом.

Префиксное кодирование

Дано отображение $\varphi\colon A \to X^+$. Является ли φ кодированием?

Префиксное кодирование

Дано отображение $\varphi\colon A \to X^+$. Является ли φ кодированием?

Очевидным необходимым условием является взаимная однозначность φ.

Префиксное кодирование

Дано отображение $\varphi\colon A \to X^+$. Является ли φ кодированием?

Очевидным необходимым условием является взаимная однозначность φ. Но взаимная однозначность φ еще недостаточна для взаимной однозначности $\bar{\varphi}$.

Префиксное кодирование

Дано отображение $\varphi\colon A \to X^+$. Является ли φ кодированием?

Очевидным необходимым условием является взаимная однозначность φ. Но взаимная однозначность φ еще недостаточна для взаимной однозначности $\bar{\varphi}$. Например, если $A = \{a, b\}$, $X = \{1\}$, $\varphi(a) = 1$, $\varphi(b) = 11$, то φ взаимно однозначно, а $\bar{\varphi}$ нет: ясно, что $\bar{\varphi}(ab) = 111 = \bar{\varphi}(ba)$.

Префиксное кодирование

Дано отображение $\varphi \colon A \to X^+$. Является ли φ кодированием?

Очевидным необходимым условием является взаимная однозначность φ. Но взаимная однозначность φ еще недостаточна для взаимной однозначности $\bar{\varphi}$. Например, если $A = \{a, b\}$, $X = \{1\}$, $\varphi(a) = 1$, $\varphi(b) = 11$, то φ взаимно однозначно, а $\bar{\varphi}$ нет: ясно, что $\bar{\varphi}(ab) = 111 = \bar{\varphi}(ba)$.

Слово $x \in X^+$ называется префиксом слова $y \in X^+$, если y можно представить, как $y = xz$ для некоторого $z \in X^+$.

Префиксное кодирование

Дано отображение $\varphi\colon A \to X^+$. Является ли φ кодированием?

Очевидным необходимым условием является взаимная однозначность φ. Но взаимная однозначность φ еще недостаточна для взаимной однозначности $\bar{\varphi}$. Например, если $A = \{a, b\}$, $X = \{1\}$, $\varphi(a) = 1$, $\varphi(b) = 11$, то φ взаимно однозначно, а $\bar{\varphi}$ нет: ясно, что $\bar{\varphi}(ab) = 111 = \bar{\varphi}(ba)$.

Слово $x \in X^+$ называется префиксом слова $y \in X^+$, если y можно представить, как $y = xz$ для некоторого $z \in X^+$. Множество слов $C \subset X^+$ называется префиксным кодом, если ни одно слово из C не является префиксом другого слова из C.

Префиксное кодирование

Дано отображение $\varphi\colon A \to X^+$. Является ли φ кодированием?

Очевидным необходимым условием является взаимная однозначность φ. Но взаимная однозначность φ еще недостаточна для взаимной однозначности $\bar{\varphi}$. Например, если $A = \{a, b\}$, $X = \{1\}$, $\varphi(a) = 1$, $\varphi(b) = 11$, то φ взаимно однозначно, а $\bar{\varphi}$ нет: ясно, что $\bar{\varphi}(ab) = 111 = \bar{\varphi}(ba)$.

Слово $x \in X^+$ называется префиксом слова $y \in X^+$, если y можно представить, как $y = xz$ для некоторого $z \in X^+$. Множество слов $C \subset X^+$ называется префиксным кодом, если ни одно слово из C не является префиксом другого слова из C.

Взаимно однозначное отображение $\varphi\colon A \to X^+$ называется префиксным кодированием, если $\varphi(A)$ – префиксный код.

Декодирование «на лету»

Несложно понять, что префиксное кодирование действительно является кодированием в определенном выше смысле.

Декодирование «на лету»

Несложно понять, что префиксное кодирование действительно является кодированием в определенном выше смысле.

Более того, декодирование, т.е. восстановление по слову $y \in \bar{\varphi}(A^+)$ того текста $w \in A^+$, для которого $\bar{\varphi}(w) = y$, в случае префиксного кодирования происходит «на лету», с той же скоростью, с какой поступают кодовые буквы.

Декодирование «на лету»

Несложно понять, что префиксное кодирование действительно является кодированием в определенном выше смысле.

Более того, декодирование, т.е. восстановление по слову $y \in \bar{\varphi}(A^+)$ того текста $w \in A^+$, для которого $\bar{\varphi}(w) = y$, в случае префиксного кодирования происходит «на лету», с той же скоростью, с какой поступают кодовые буквы.

Действительно, мы просто читаем слово y слева направо, пока не прочтем префикс x, являющийся словом из $C := \varphi(A)$.

Декодирование «на лету»

Несложно понять, что префиксное кодирование действительно является кодированием в определенном выше смысле.

Более того, декодирование, т.е. восстановление по слову $y \in \bar{\varphi}(A^+)$ того текста $w \in A^+$, для которого $\bar{\varphi}(w) = y$, в случае префиксного кодирования происходит «на лету», с той же скоростью, с какой поступают кодовые буквы.

Действительно, мы просто читаем слово y слева направо, пока не прочтем префикс x, являющийся словом из $C := \varphi(A)$. Этот префикс x единствен в силу определения префиксного кода: если бы y имело два разных префикса из C, то более короткий был бы префиксом более длинного, что невозможно.

Декодирование «на лету»

Несложно понять, что префиксное кодирование действительно является кодированием в определенном выше смысле.

Более того, декодирование, т.е. восстановление по слову $y \in \bar{\varphi}(A^+)$ того текста $w \in A^+$, для которого $\bar{\varphi}(w) = y$, в случае префиксного кодирования происходит «на лету», с той же скоростью, с какой поступают кодовые буквы.

Действительно, мы просто читаем слово y слева направо, пока не прочтем префикс x, являющийся словом из $C := \varphi(A)$. Этот префикс x единствен в силу определения префиксного кода: если бы y имело два разных префикса из C, то более короткий был бы префиксом более длинного, что невозможно. Если $x = \varphi(a)$, то a – первый знак текста w; теперь удаляем x из y и применяем ту же процедуру к оставшемуся слову.

Декодирование «на лету»

Несложно понять, что префиксное кодирование действительно является кодированием в определенном выше смысле.

Более того, декодирование, т.е. восстановление по слову $y \in \bar{\varphi}(A^+)$ того текста $w \in A^+$, для которого $\bar{\varphi}(w) = y$, в случае префиксного кодирования происходит «на лету», с той же скоростью, с какой поступают кодовые буквы.

Действительно, мы просто читаем слово y слева направо, пока не прочтем префикс x, являющийся словом из $C := \varphi(A)$. Этот префикс x единствен в силу определения префиксного кода: если бы y имело два разных префикса из C, то более короткий был бы префиксом более длинного, что невозможно. Если $x = \varphi(a)$, то a – первый знак текста w; теперь удаляем x из y и применяем ту же процедуру к оставшемуся слову.

Описанный алгоритм изящно реализуется конечным автоматом (но эту тему мы не будем развивать).

Однозначность vs декодирование «на лету»

Для сравнения: если $A = \{a, b\}$, $X = \{0, 1\}$, $\varphi(a) = 0$, $\varphi(b) = 01$, то можно проверить, что φ будет кодированием. (**Контрольный вопрос**: как доказать это быстрее всего?)

Однозначность vs декодирование «на лету»

Для сравнения: если $A = \{a, b\}$, $X = \{0, 1\}$, $\varphi(a) = 0$, $\varphi(b) = 01$, то можно проверить, что φ будет кодированием. (**Контрольный вопрос**: как доказать это быстрее всего?)

Однако декодировать слева направо «на лету» не получится. Скажем, для $y = 001$ первый бит 0 – это $\varphi(a)$, но мы не можем это определить, пока не прочитаем второй бит.

Однозначность vs декодирование «на лету»

Для сравнения: если $A = \{a, b\}$, $X = \{0, 1\}$, $\varphi(a) = 0$, $\varphi(b) = 01$, то можно проверить, что φ будет кодированием. (**Контрольный вопрос**: как доказать это быстрее всего?)

Однако декодировать слева направо «на лету» не получится. Скажем, для $y = 001$ первый бит 0 – это $\varphi(a)$, но мы не можем это определить, пока не прочитаем второй бит.

Итак, префиксность – достаточное условие для возможности однозначного декодирования, но не необходимое.

Однозначность vs декодирование «на лету»

Для сравнения: если $A = \{a, b\}$, $X = \{0, 1\}$, $\varphi(a) = 0$, $\varphi(b) = 01$, то можно проверить, что φ будет кодированием. (**Контрольный вопрос**: как доказать это быстрее всего?)

Однако декодировать слева направо «на лету» не получится. Скажем, для $y = 001$ первый бит 0 – это $\varphi(a)$, но мы не можем это определить, пока не прочитаем второй бит.

Итак, префиксность – достаточное условие для возможности однозначного декодирования, но не необходимое.

Необходимое и достаточное условие возможности однозначного декодирования известно (алгоритм Сардинаса–Паттерсона, 1953), но мы не будем его обсуждать, поскольку докажем, что для нужд задач сжатия без потери информации префиксных кодов в некотором (точном) смысле хватает.

Теорема Крафта: необходимость

Теорема (Крафт, 1949, магистерская диссертация в МИТ)
Пусть $|X| = d$, $\ell_1, \ldots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \ldots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Теорема Крафта: необходимость

Теорема (Крафт, 1949, магистерская диссертация в МИТ) Пусть $|X| = d$, $\ell_1, \ldots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \ldots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Необходимость. Обозначим через k_j число слов длины j в C.

Теорема Крафта: необходимость

Теорема (Крафт, 1949, магистерская диссертация в МИТ)
Пусть $|X| = d$, $\ell_1, \ldots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \ldots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Необходимость. Обозначим через k_j число слов длины j в C. Ясно, что $k_1 \leqslant d$.

Теорема Крафта: необходимость

Теорема (Крафт, 1949, магистерская диссертация в МИТ) Пусть $|X| = d$, $\ell_1, \ldots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \ldots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Необходимость. Обозначим через k_j число слов длины j в C. Ясно, что $k_1 \leqslant d$. Так как C – префиксный код, $\leqslant d - k_1$ слов длины 1 могут быть префиксами более длинных слов из C.

Теорема Крафта: необходимость

Теорема (Крафт, 1949, магистерская диссертация в МИТ)
Пусть $|X| = d$, $\ell_1, \ldots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \ldots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Необходимость. Обозначим через k_j число слов длины j в C. Ясно, что $k_1 \leqslant d$. Так как C – префиксный код, $\leqslant d - k_1$ слов длины 1 могут быть префиксами более длинных слов из C. Отсюда $k_2 \leqslant (d - k_1)d = d^2 - k_1 d$.

Теорема Крафта: необходимость

Теорема (Крафт, 1949, магистерская диссертация в МИТ)
Пусть $|X| = d$, $\ell_1, \ldots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \ldots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Необходимость. Обозначим через k_j число слов длины j в C. Ясно, что $k_1 \leqslant d$. Так как C – префиксный код, $\leqslant d - k_1$ слов длины 1 могут быть префиксами более длинных слов из C. Отсюда $k_2 \leqslant (d - k_1)d = d^2 - k_1 d$. Аналогично, максимум $(d - k_1)d - k_2$ слов длины 2 служат префиксами более длинных слов из C, откуда $k_3 \leqslant ((d - k_1)d - k_2)d = d^3 - k_1 d^2 - k_2 d$.

Теорема Крафта: необходимость

Теорема (Крафт, 1949, магистерская диссертация в МИТ)
Пусть $|X| = d$, $\ell_1, \ldots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \ldots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Необходимость. Обозначим через k_j число слов длины j в C. Ясно, что $k_1 \leqslant d$. Так как C – префиксный код, $\leqslant d - k_1$ слов длины 1 могут быть префиксами более длинных слов из C. Отсюда $k_2 \leqslant (d - k_1)d = d^2 - k_1 d$. Аналогично, максимум $(d - k_1)d - k_2$ слов длины 2 служат префиксами более длинных слов из C, откуда $k_3 \leqslant ((d - k_1)d - k_2)d = d^3 - k_1 d^2 - k_2 d$.

Теорема Крафта: необходимость

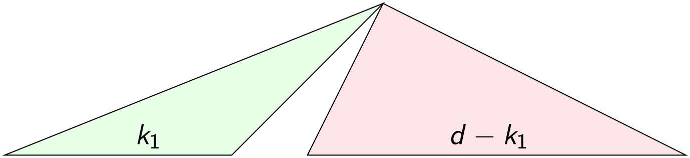
Теорема (Крафт, 1949, магистерская диссертация в МИТ)
Пусть $|X| = d$, $\ell_1, \dots, \ell_n$ – последовательность натуральных чисел (не обязательно различных). Префиксный код $C \subset X^+$ из n слов с длинами $\ell_1, \dots, \ell_n$ существует тогда и только тогда, когда для чисел $d, \ell_1, \dots, \ell_n$ выполняется неравенство Крафта:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

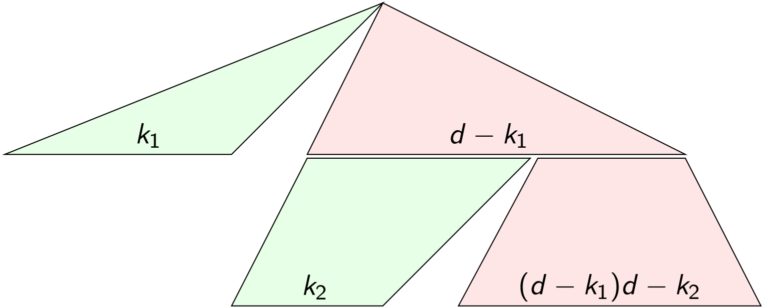
Необходимость. Обозначим через k_j число слов длины j в C. Ясно, что $k_1 \leqslant d$. Так как C – префиксный код, $\leqslant d - k_1$ слов длины 1 могут быть префиксами более длинных слов из C. Отсюда $k_2 \leqslant (d - k_1)d = d^2 - k_1 d$. Аналогично, максимум $(d - k_1)d - k_2$ слов длины 2 служат префиксами более длинных слов из C, откуда $k_3 \leqslant ((d - k_1)d - k_2)d = d^3 - k_1 d^2 - k_2 d$.

Этот аргумент становится наглядным, если визуализировать слова из X^+ как пути из корня в полном d-ичном дереве.

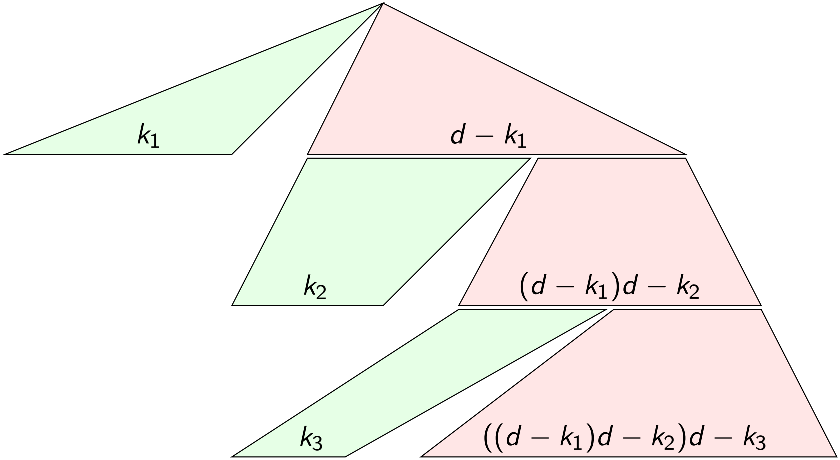
Теорема Крафта: необходимость, визуализация



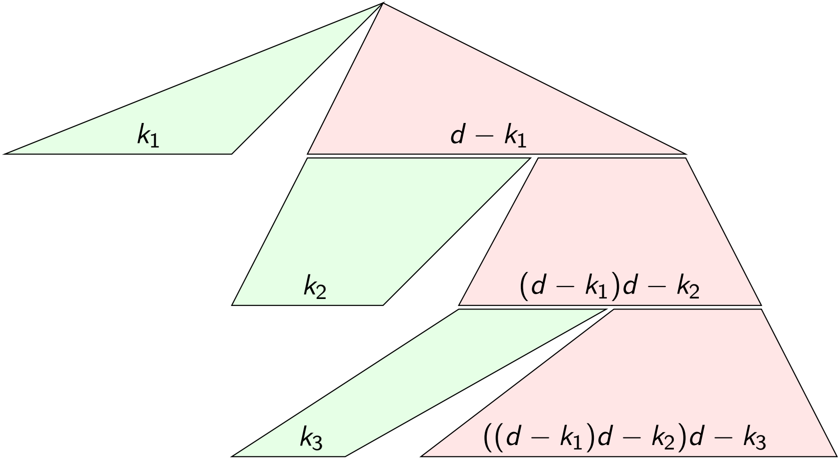
Теорема Крафта: необходимость, визуализация



Теорема Крафта: необходимость, визуализация



Теорема Крафта: необходимость, визуализация



и так далее

Теорема Крафта: необходимость, окончание

Продолжая в том же духе, получаем, что для любого s выполняется неравенство

$$k_s \leqslant d^s - k_1 d^{s-1} - \cdots - k_{s-1} d,$$

Теорема Крафта: необходимость, окончание

Продолжая в том же духе, получаем, что для любого s выполняется неравенство

$$k_s \leqslant d^s - k_1 d^{s-1} - \cdots - k_{s-1} d,$$

которое можно переписать в виде

$$k_1 d^{s-1} + \cdots + k_{s-1} d + k_s \leqslant d^s.$$

Теорема Крафта: необходимость, окончание

Продолжая в том же духе, получаем, что для любого s выполняется неравенство

$$k_s \leqslant d^s - k_1 d^{s-1} - \dots - k_{s-1} d,$$

которое можно переписать в виде

$$k_1 d^{s-1} + \dots + k_{s-1} d + k_s \leqslant d^s.$$

Поделив каждый член на d^s, получим $\sum_{j=1}^{s} k_j d^{-j} \leqslant 1 \quad (\star)$.

Теорема Крафта: необходимость, окончание

Продолжая в том же духе, получаем, что для любого s выполняется неравенство

$$k_s \leqslant d^s - k_1 d^{s-1} - \cdots - k_{s-1} d,$$

которое можно переписать в виде

$$k_1 d^{s-1} + \cdots + k_{s-1} d + k_s \leqslant d^s.$$

Поделив каждый член на d^s, получим $\sum_{j=1}^{s} k_j d^{-j} \leqslant 1 \quad (\star)$.

Вспомним: k_j – число слов длины j в C, т.е. число появлений j в последовательности $\ell_1, \ldots, \ell_n$. Пусть $J := \max_i \{\ell_i\}$.

Теорема Крафта: необходимость, окончание

Продолжая в том же духе, получаем, что для любого s выполняется неравенство

$$k_s \leqslant d^s - k_1 d^{s-1} - \cdots - k_{s-1} d,$$

которое можно переписать в виде

$$k_1 d^{s-1} + \cdots + k_{s-1} d + k_s \leqslant d^s.$$

Поделив каждый член на d^s, получим $\sum_{j=1}^{s} k_j d^{-j} \leqslant 1 \quad (\star)$.

Вспомним: k_j – число слов длины j в C, т.е. число появлений j в последовательности $\ell_1, \ldots, \ell_n$. Пусть $J := \max_i \{\ell_i\}$.

Несложно понять, что $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$ – левая часть получается из правой группировкой одинаковых слагаемых.

Теорема Крафта: необходимость, окончание

Продолжая в том же духе, получаем, что для любого s выполняется неравенство

$$k_s \leqslant d^s - k_1 d^{s-1} - \cdots - k_{s-1} d,$$

которое можно переписать в виде

$$k_1 d^{s-1} + \cdots + k_{s-1} d + k_s \leqslant d^s.$$

Поделив каждый член на d^s, получим $\sum_{j=1}^{s} k_j d^{-j} \leqslant 1 \quad (\star)$.

Вспомним: k_j – число слов длины j в C, т.е. число появлений j в последовательности $\ell_1, \ldots, \ell_n$. Пусть $J := \max_i \{\ell_i\}$.

Несложно понять, что $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$ – левая часть получается из правой группировкой одинаковых слагаемых.

Применив $(\star)$ при $s = J$, получаем неравенство Крафта

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \dots, \ell_n$ удовлетворяют неравенству

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \ldots, \ell_n$ удовлетворяют неравенству
$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$
«Развернем» аргументы из доказательства необходимости.

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \ldots, \ell_n$ удовлетворяют неравенству

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

«Развернем» аргументы из доказательства необходимости.
Пусть k_j – число появлений j в последовательности $\{\ell_i\}$
и $J := \max_i\{\ell_i\}$.

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \ldots, \ell_n$ удовлетворяют неравенству

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

«Развернем» аргументы из доказательства необходимости.
Пусть k_j – число появлений j в последовательности $\{\ell_i\}$
и $J := \max_i\{\ell_i\}$. Тогда, как и выше, $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$.

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \ldots, \ell_n$ удовлетворяют неравенству

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

«Развернем» аргументы из доказательства необходимости.
Пусть k_j – число появлений j в последовательности $\{\ell_i\}$
и $J := \max_i\{\ell_i\}$. Тогда, как и выше, $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$.
Отсюда неравенство Крафта перепишется как $\sum_{j=1}^{J} k_j d^{-j} \leqslant 1$.

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \ldots, \ell_n$ удовлетворяют неравенству

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

«Развернем» аргументы из доказательства необходимости.
Пусть k_j – число появлений j в последовательности $\{\ell_i\}$
и $J := \max_i\{\ell_i\}$. Тогда, как и выше, $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$.
Отсюда неравенство Крафта перепишется как $\sum_{j=1}^{J} k_j d^{-j} \leqslant 1$.
Но тогда неравенство $\sum_{j=1}^{s} k_j d^{-j} \leqslant 1 \quad (\star)$ верно при любом s.

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \ldots, \ell_n$ удовлетворяют неравенству

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

«Развернем» аргументы из доказательства необходимости.
Пусть k_j – число появлений j в последовательности $\{\ell_i\}$
и $J := \max_i\{\ell_i\}$. Тогда, как и выше, $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$.
Отсюда неравенство Крафта перепишется как $\sum_{j=1}^{J} k_j d^{-j} \leqslant 1$.
Но тогда неравенство $\sum_{j=1}^{s} k_j d^{-j} \leqslant 1 \quad (\star)$ верно при любом s.
Домножая $(\star)$ на d^s, получим

$$k_1 d^{s-1} + \cdots + k_{s-1} d + k_s \leqslant d^s,$$

Теорема Крафта: достаточность

Теперь дано, что числа $d, \ell_1, \ldots, \ell_n$ удовлетворяют неравенству

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1.$$

«Развернем» аргументы из доказательства необходимости.
Пусть k_j – число появлений j в последовательности $\{\ell_i\}$
и $J := \max_i\{\ell_i\}$. Тогда, как и выше, $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$.
Отсюда неравенство Крафта перепишется как $\sum_{j=1}^{J} k_j d^{-j} \leqslant 1$.
Но тогда неравенство $\sum_{j=1}^{s} k_j d^{-j} \leqslant 1 \quad (\star)$ верно при любом s.
Домножая $(\star)$ на d^s, получим

$$k_1 d^{s-1} + \cdots + k_{s-1} d + k_s \leqslant d^s,$$

откуда

$$k_s \leqslant d^s - k_1 d^{s-1} - \cdots - k_{s-1} d = (\ldots((d-k_1)d-k_2)d - \cdots - k_{s-1})d.$$

Теорема Крафта: достаточность, окончание

Итак, неравенство Крафта обеспечивает выполнение системы

$$\begin{aligned} k_1 &\leqslant d, \\ k_2 &\leqslant (d - k_1)d, \\ k_3 &\leqslant ((d - k_1)d - k_2)d, \\ &\dots\dots\dots\dots\dots\dots \\ k_s &\leqslant (\dots((d - k_1)d - k_2)d - \dots - k_{s-1})d, \\ &\dots\dots\dots\dots\dots\dots \end{aligned} \tag{$\dagger$}$$

Теорема Крафта: достаточность, окончание

Итак, неравенство Крафта обеспечивает выполнение системы

$$\begin{aligned}
k_1 &\leqslant d, \\
k_2 &\leqslant (d - k_1)d, \\
k_3 &\leqslant ((d - k_1)d - k_2)d, \qquad (\dagger) \\
&\dots\dots\dots\dots\dots\dots\dots \\
k_s &\leqslant (\dots((d - k_1)d - k_2)d - \dots - k_{s-1})d, \\
&\dots\dots\dots\dots\dots\dots\dots\dots\dots
\end{aligned}$$

Теперь строим множество слов $C \subset X^+$ следующим образом: берем k_1 букв,

Теорема Крафта: достаточность, окончание

Итак, неравенство Крафта обеспечивает выполнение системы

$$
\begin{aligned}
k_1 &\leqslant d, \\
k_2 &\leqslant (d - k_1)d, \\
k_3 &\leqslant ((d - k_1)d - k_2)d, \\
&\dots\dots\dots\dots\dots\dots \\
k_s &\leqslant (\dots((d - k_1)d - k_2)d - \dots - k_{s-1})d, \\
&\dots\dots\dots\dots\dots\dots
\end{aligned}
\tag{$\dagger$}
$$

Теперь строим множество слов $C \subset X^+$ следующим образом: берем k_1 букв, потом добавляем k_2 слов длины 2, для которых выбранные буквы не являются префиксами,

Теорема Крафта: достаточность, окончание

Итак, неравенство Крафта обеспечивает выполнение системы

$$
\begin{gathered}
k_1 \leqslant d, \\
k_2 \leqslant (d - k_1)d, \\
k_3 \leqslant ((d - k_1)d - k_2)d, \\
\dots\dots\dots\dots\dots\dots\dots\dots \\
k_s \leqslant (\dots((d - k_1)d - k_2)d - \dots - k_{s-1})d, \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots
\end{gathered}
\tag{$\dagger$}
$$

Теперь строим множество слов $C \subset X^+$ следующим образом: берем k_1 букв, потом добавляем k_2 слов длины 2, для которых выбранные буквы не являются префиксами, потом – k_3 слов длины 3, для которых выбранные слова длины $\leqslant 2$ не являются префиксами,

Теорема Крафта: достаточность, окончание

Итак, неравенство Крафта обеспечивает выполнение системы

$$
\begin{gathered}
k_1 \leqslant d, \\
k_2 \leqslant (d - k_1)d, \\
k_3 \leqslant ((d - k_1)d - k_2)d, \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
k_s \leqslant (\dots((d - k_1)d - k_2)d - \dots - k_{s-1})d, \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots
\end{gathered}
\tag{$\dagger$}
$$

Теперь строим множество слов $C \subset X^+$ следующим образом: берем k_1 букв, потом добавляем k_2 слов длины 2, для которых выбранные буквы не являются префиксами, потом – k_3 слов длины 3, для которых выбранные слова длины $\leqslant 2$ не являются префиксами, ..., потом – k_s слов длины s, для которых выбранные слова длины $\leqslant s - 1$ не являются префиксами, ...

Теорема Крафта: достаточность, окончание

Итак, неравенство Крафта обеспечивает выполнение системы

$$
\begin{aligned}
k_1 &\leqslant d, \\
k_2 &\leqslant (d - k_1)d, \\
k_3 &\leqslant ((d - k_1)d - k_2)d, \qquad (\dagger) \\
&\dots\dots\dots\dots\dots\dots\dots \\
k_s &\leqslant (\dots((d - k_1)d - k_2)d - \dots - k_{s-1})d, \\
&\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots
\end{aligned}
$$

Теперь строим множество слов $C \subset X^+$ следующим образом: берем k_1 букв, потом добавляем k_2 слов длины 2, для которых выбранные буквы не являются префиксами, потом – k_3 слов длины 3, для которых выбранные слова длины $\leqslant 2$ не являются префиксами, . . . , потом – k_s слов длины s, для которых выбранные слова длины $\leqslant s - 1$ не являются префиксами, ...

В силу неравенств $(\dagger)$ каждый шаг возможен,

Теорема Крафта: достаточность, окончание

Итак, неравенство Крафта обеспечивает выполнение системы

$$\begin{aligned} k_1 &\leqslant d, \\ k_2 &\leqslant (d-k_1)d, \\ k_3 &\leqslant ((d-k_1)d-k_2)d, \\ &\dots\dots\dots\dots\dots\dots \\ k_s &\leqslant (\dots((d-k_1)d-k_2)d-\dots-k_{s-1})d, \\ &\dots\dots\dots\dots\dots\dots \end{aligned} \tag{$\dagger$}$$

Теперь строим множество слов $C \subset X^+$ следующим образом: берем k_1 букв, потом добавляем k_2 слов длины 2, для которых выбранные буквы не являются префиксами, потом – k_3 слов длины 3, для которых выбранные слова длины $\leqslant 2$ не являются префиксами, …, потом – k_s слов длины s, для которых выбранные слова длины $\leqslant s-1$ не являются префиксами, ….

В силу неравенств ($\dagger$) каждый шаг возможен, и по построению C – префиксный код из n слов с длинами $\ell_1, \dots, \ell_n$.

Теорема Макмиллана

Наше доказательство неравенства Крафта существенно использовало префиксность рассматриваемого кода.

Теорема Макмиллана

Наше доказательство неравенства Крафта существенно использовало префиксность рассматриваемого кода.
В действительности, однако, неравенство верно для всех кодов!

Теорема Макмиллана

Наше доказательство неравенства Крафта существенно использовало префиксность рассматриваемого кода.
В действительности, однако, неравенство верно для всех кодов!

Теорема (Макмиллан, 1956)
Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.
Тогда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1. \qquad (*)$$

Теорема Макмиллана

Наше доказательство неравенства Крафта существенно использовало префиксность рассматриваемого кода.
В действительности, однако, неравенство верно для всех кодов!

Теорема (Макмиллан, 1956)
Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.
Тогда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1. \qquad (*)$$

С учетом этой теоремы, будем далее именовать неравенство $(*)$ неравенством Крафта–Макмиллана.

Теорема Макмиллана

Наше доказательство неравенства Крафта существенно использовало префиксность рассматриваемого кода.
В действительности, однако, неравенство верно для всех кодов!

Теорема (Макмиллан, 1956)
Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.
Тогда для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1. \qquad (*)$$

С учетом этой теоремы, будем далее именовать неравенство $(*)$ неравенством Крафта–Макмиллана.

Обратная теорема к теореме Макмиллана тоже верна, но мы ее не формулируем потому что она содержится в теореме Крафта.

Теорема Макмиллана: доказательство

Используем те же обозначения, что и в доказательстве теоремы Крафта: пусть k_j – число слов длины j в C и $J := \max_i\{\ell_i\}$.

Теорема Макмиллана: доказательство

Используем те же обозначения, что и в доказательстве теоремы Крафта: пусть k_j – число слов длины j в C и $J := \max_i\{\ell_i\}$. Тогда $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$.

Теорема Макмиллана: доказательство

Используем те же обозначения, что и в доказательстве теоремы Крафта: пусть k_j – число слов длины j в C и $J := \max_i\{\ell_i\}$. Тогда $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$. Рассмотрим многочлен

$$Q(x) := \sum_{j=1}^{J} k_j x^j.$$

Теорема Макмиллана: доказательство

Используем те же обозначения, что и в доказательстве теоремы Крафта: пусть k_j – число слов длины j в C и $J := \max_i\{\ell_i\}$. Тогда $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$. Рассмотрим многочлен

$$Q(x) := \sum_{j=1}^{J} k_j x^j.$$

Достаточно доказать, что многочлен $Q(x) - 1$ не имеет корней в интервале $0 < x < d^{-1}$.

Теорема Макмиллана: доказательство

Используем те же обозначения, что и в доказательстве теоремы Крафта: пусть k_j – число слов длины j в C и $J := \max_i\{\ell_i\}$. Тогда $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$. Рассмотрим многочлен

$$Q(x) := \sum_{j=1}^{J} k_j x^j.$$

Достаточно доказать, что многочлен $Q(x) - 1$ не имеет корней в интервале $0 < x < d^{-1}$. Действительно, $Q(0) - 1 = -1$, а поскольку функция $Q(x) - 1$ непрерывна, при переходе от отрицательных значений к положительным она должна принять нулевое значение.

Теорема Макмиллана: доказательство

Используем те же обозначения, что и в доказательстве теоремы Крафта: пусть k_j – число слов длины j в C и $J := \max_i\{\ell_i\}$. Тогда $\sum_{j=1}^{J} k_j d^{-j} = \sum_{i=1}^{n} d^{-\ell_i}$. Рассмотрим многочлен

$$Q(x) := \sum_{j=1}^{J} k_j x^j.$$

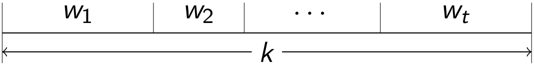
Достаточно доказать, что многочлен $Q(x) - 1$ не имеет корней в интервале $0 < x < d^{-1}$. Действительно, $Q(0) - 1 = -1$, а поскольку функция $Q(x) - 1$ непрерывна, при переходе от отрицательных значений к положительным она должна принять нулевое значение. Если корней в интервале $0 < x < d^{-1}$ нет, то $Q(x) - 1 < 0$ при всех $0 \leqslant x < d^{-1}$, откуда $Q(d^{-1}) - 1 \leqslant 0$, т.е. $\sum_{j=1}^{J} k_j d^{-j} = Q(d^{-1}) \leqslant 1$.

Теорема Макмиллана: доказательство, продолжение

Обозначим через $N(k)$ число произведений слов из кода C, длина которых (как слов из X^+) равна k.

Теорема Макмиллана: доказательство, продолжение

Обозначим через $N(k)$ число произведений слов из кода C, длина которых (как слов из X^+) равна k.



$w_1, w_2, \ldots, w_t \in C$

Теорема Макмиллана: доказательство, продолжение

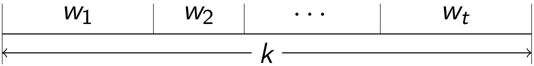
Обозначим через $N(k)$ число произведений слов из кода C, длина которых (как слов из X^+) равна k.

$$\underbrace{w_1 \mid w_2 \mid \cdots \mid w_t}_{k} \qquad w_1, w_2, \ldots, w_t \in C$$

Поскольку C – код, разные произведения слов из C остаются разными после «раскрытия скобок», т.е. после их перезаписи в виде слов из X^+.

Теорема Макмиллана: доказательство, продолжение

Обозначим через $N(k)$ число произведений слов из кода C, длина которых (как слов из X^+) равна k.



$w_1, w_2, \ldots, w_t \in C$

Поскольку C – код, разные произведения слов из C остаются разными после «раскрытия скобок», т.е. после их перезаписи в виде слов из X^+. Поэтому $N(k) \leqslant d^k$ (правая часть – просто число всех слов длины k в X^+).

Теорема Макмиллана: доказательство, продолжение

Обозначим через $N(k)$ число произведений слов из кода C, длина которых (как слов из X^+) равна k.

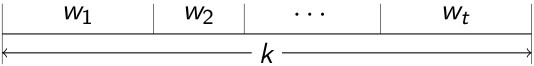
$$\underbrace{w_1 \mid w_2 \mid \cdots \mid w_t}_{k} \qquad w_1, w_2, \ldots, w_t \in C$$

Поскольку C – код, разные произведения слов из C остаются разными после «раскрытия скобок», т.е. после их перезаписи в виде слов из X^+. Поэтому $N(k) \leqslant d^k$ (правая часть – просто число всех слов длины k в X^+). Рассмотрим степенной ряд

$$F(x) = 1 + N(1)x + N(2)x^2 + \cdots + N(k)x^k + \cdots .$$

Теорема Макмиллана: доказательство, продолжение

Обозначим через $N(k)$ число произведений слов из кода C, длина которых (как слов из X^+) равна k.



$w_1, w_2, \dots, w_t \in C$

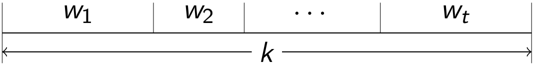
Поскольку C – код, разные произведения слов из C остаются разными после «раскрытия скобок», т.е. после их перезаписи в виде слов из X^+. Поэтому $N(k) \leqslant d^k$ (правая часть – просто число всех слов длины k в X^+). Рассмотрим степенной ряд

$$F(x) = 1 + N(1)x + N(2)x^2 + \cdots + N(k)x^k + \cdots .$$

Поскольку $N(k) \leqslant d^k$, ряд $F(x)$ мажорируется суммой членов геометрической прогрессии $1 + dx + d^2x^2 + \cdots + d^kx^k + \cdots$, а стало быть, этот ряд сходится в интервале $-d^{-1} < x < d^{-1}$, где сходится указанная сумма.

Теорема Макмиллана: доказательство, продолжение

Обозначим через $N(k)$ число произведений слов из кода C, длина которых (как слов из X^+) равна k.



$w_1, w_2, \ldots, w_t \in C$

Поскольку C – код, разные произведения слов из C остаются разными после «раскрытия скобок», т.е. после их перезаписи в виде слов из X^+. Поэтому $N(k) \leqslant d^k$ (правая часть – просто число всех слов длины k в X^+). Рассмотрим степенной ряд

$$F(x) = 1 + N(1)x + N(2)x^2 + \cdots + N(k)x^k + \cdots .$$

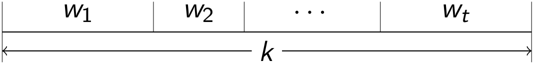
Поскольку $N(k) \leqslant d^k$, ряд $F(x)$ мажорируется суммой членов геометрической прогрессии $1 + dx + d^2x^2 + \cdots + d^kx^k + \cdots$, а стало быть, этот ряд сходится в интервале $-d^{-1} < x < d^{-1}$, где сходится указанная сумма. Заключаем, что функция $F(x)$ определена при всех x из интервала $-d^{-1} < x < d^{-1}$.

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j.

Теорема Макмиллана: доказательство, окончание

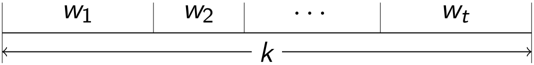
Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j.



w_1 – слово длины j

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.



w_1 – слово длины j

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$.

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.)

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x)$$

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x) = \sum_{k=0}^{\infty} N(k)x^k \cdot \sum_{j=1}^{J} k_j x^j$$

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x) = \sum_{k=0}^{\infty} N(k)x^k \cdot \sum_{j=1}^{J} k_j x^j = \sum_{k=1}^{\infty} \left(\sum_{j=1}^{J} k_j N(k-j) \right) x^k$$

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x) = \sum_{k=0}^{\infty} N(k)x^k \cdot \sum_{j=1}^{J} k_j x^j = \sum_{k=1}^{\infty} \left(\sum_{j=1}^{J} k_j N(k-j) \right) x^k$$
$$= \sum_{k=1}^{\infty} N(k)x^k$$

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x) = \sum_{k=0}^{\infty} N(k)x^k \cdot \sum_{j=1}^{J} k_j x^j = \sum_{k=1}^{\infty} \left(\sum_{j=1}^{J} k_j N(k-j) \right) x^k$$

$$= \sum_{k=1}^{\infty} N(k)x^k = F(x) - 1.$$

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x) = \sum_{k=0}^{\infty} N(k)x^k \cdot \sum_{j=1}^{J} k_j x^j = \sum_{k=1}^{\infty} \left(\sum_{j=1}^{J} k_j N(k-j) \right) x^k$$
$$= \sum_{k=1}^{\infty} N(k)x^k = F(x) - 1.$$

Отсюда $F(x)(1 - Q(x)) = 1$.

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x) = \sum_{k=0}^{\infty} N(k)x^k \cdot \sum_{j=1}^{J} k_j x^j = \sum_{k=1}^{\infty} \left(\sum_{j=1}^{J} k_j N(k-j) \right) x^k$$
$$= \sum_{k=1}^{\infty} N(k)x^k = F(x) - 1.$$

Отсюда $F(x)(1 - Q(x)) = 1$. Функция $F(x)$ определена на интервале $-d^{-1} < x < d^{-1}$; значит, $1 - Q(x) \neq 0$ на этом интервале (и, в частности, при $0 < x < d^{-1}$).

Теорема Макмиллана: доказательство, окончание

Обозначим через $P_j(k)$ множество всех тех произведений слов из C, у которых длина (как у слов из X^+) равна k и первый сомножитель – слово длины j. Тогда $|P_j(k)| = k_j N(k-j)$.

Множества $P_j(k)$ при разных j не пересекаются (ведь C – код!), откуда $N(k) = k_1 N(k-1) + k_2 N(k-2) + \cdots + k_J N(k-J)$. (Здесь мы положили $N(0) = 1$ и $N(k) = 0$ при $k < 0$, чтобы последнее равенство имело смысл для всех k.) Теперь имеем

$$F(x)Q(x) = \sum_{k=0}^{\infty} N(k)x^k \cdot \sum_{j=1}^{J} k_j x^j = \sum_{k=1}^{\infty} \left(\sum_{j=1}^{J} k_j N(k-j) \right) x^k$$
$$= \sum_{k=1}^{\infty} N(k)x^k = F(x) - 1.$$

Отсюда $F(x)(1 - Q(x)) = 1$. Функция $F(x)$ определена на интервале $-d^{-1} < x < d^{-1}$; значит, $1 - Q(x) \neq 0$ на этом интервале (и, в частности, при $0 < x < d^{-1}$). Именно это оставалось доказать.

Важное следствие

Сопоставляя теорему Макмиллана и теорему Крафта, получаем важное

Следствие. Если существует какой-то код из n слов с длинами $\ell_1, \ldots, \ell_n$ над алфавитом X, то над X имеется и префиксный код с теми же параметрами.

Важное следствие

Сопоставляя теорему Макмиллана и теорему Крафта, получаем важное

Следствие. Если существует какой-то код из n слов с длинами $\ell_1, \ldots, \ell_n$ над алфавитом X, то над X имеется и префиксный код с теми же параметрами.

Действительно, если $|X| = d$ и над X есть код из n слов с длинами $\ell_1, \ldots, \ell_n$, то по теореме Макмиллана для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта–Макмиллана:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1. \qquad (*)$$

Важное следствие

Сопоставляя теорему Макмиллана и теорему Крафта, получаем важное

Следствие. Если существует какой-то код из n слов с длинами $\ell_1, \ldots, \ell_n$ над алфавитом X, то над X имеется и префиксный код с теми же параметрами.

Действительно, если $|X| = d$ и над X есть код из n слов с длинами $\ell_1, \ldots, \ell_n$, то по теореме Макмиллана для чисел $d, \ell_1, \ldots, \ell_n$ выполняется неравенство Крафта–Макмиллана:

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant 1. \qquad (*)$$

С другой стороны, по теореме Крафта неравенство $(*)$ гарантирует существование над X префиксного кода из n слов с длинами $\ell_1, \ldots, \ell_n$.

Степень сжатия

Пусть $A = \{a_1, \ldots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Степень сжатия

Пусть $A = \{a_1, \dots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.

Степень сжатия

Пусть $A = \{a_1, \ldots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.
Тогда $L := \sum_{i=1}^{n} p_i \ell_i$ – средняя длина кодового слова (в расчете на один знак текста).

Степень сжатия

Пусть $A = \{a_1, \dots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.
Тогда $L := \sum_{i=1}^{n} p_i\ell_i$ – средняя длина кодового слова (в расчете на один знак текста).
Именно этот параметр определяет степень сжатия: чем он меньше, тем кодирование лучше.

Степень сжатия

Пусть $A = \{a_1, \ldots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.
Тогда $L := \sum_{i=1}^{n} p_i \ell_i$ – средняя длина кодового слова (в расчете на один знак текста).
Именно этот параметр определяет степень сжатия:
чем он меньше, тем кодирование лучше.

Упражнение. Первой фразой, которую Морзе передал по телеграфу 24 мая 1844 года, была цитата из Книги Чисел: **WHAT HATH GOD WROUGHT** (Вот что творит Бог).

Степень сжатия

Пусть $A = \{a_1, \dots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.
Тогда $L := \sum_{i=1}^{n} p_i \ell_i$ – средняя длина кодового слова (в расчете на один знак текста).
Именно этот параметр определяет степень сжатия: чем он меньше, тем кодирование лучше.

Упражнение. Первой фразой, которую Морзе передал по телеграфу 24 мая 1844 года, была цитата из Книги Чисел: **WHAT HATH GOD WROUGHT** (Вот что творит Бог). Подсчитайте среднюю длину кодового слова при кодировании этой фразы кодом Морзе.

Степень сжатия

Пусть $A = \{a_1, \ldots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.
Тогда $L := \sum_{i=1}^{n} p_i \ell_i$ – средняя длина кодового слова (в расчете на один знак текста).
Именно этот параметр определяет степень сжатия: чем он меньше, тем кодирование лучше.

Упражнение. Первой фразой, которую Морзе передал по телеграфу 24 мая 1844 года, была цитата из Книги Чисел: **WHAT HATH GOD WROUGHT** (Вот что творит Бог). Подсчитайте среднюю длину кодового слова при кодировании этой фразы кодом Морзе. Учтите, что код Морзе троичный: кодовый алфавит состоит из точки, тире и паузы.

Степень сжатия

Пусть $A = \{a_1, \dots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi\colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.
Тогда $L := \sum_{i=1}^{n} p_i \ell_i$ – средняя длина кодового слова (в расчете на один знак текста).
Именно этот параметр определяет степень сжатия:
чем он меньше, тем кодирование лучше.

Упражнение. Первой фразой, которую Морзе передал по телеграфу 24 мая 1844 года, была цитата из Книги Чисел: **WHAT HATH GOD WROUGHT** (Вот что творит Бог). Подсчитайте среднюю длину кодового слова при кодировании этой фразы кодом Морзе. Учтите, что код Морзе троичный: кодовый алфавит состоит из точки, тире и паузы. Скажем, код **E** – это точка и пауза.

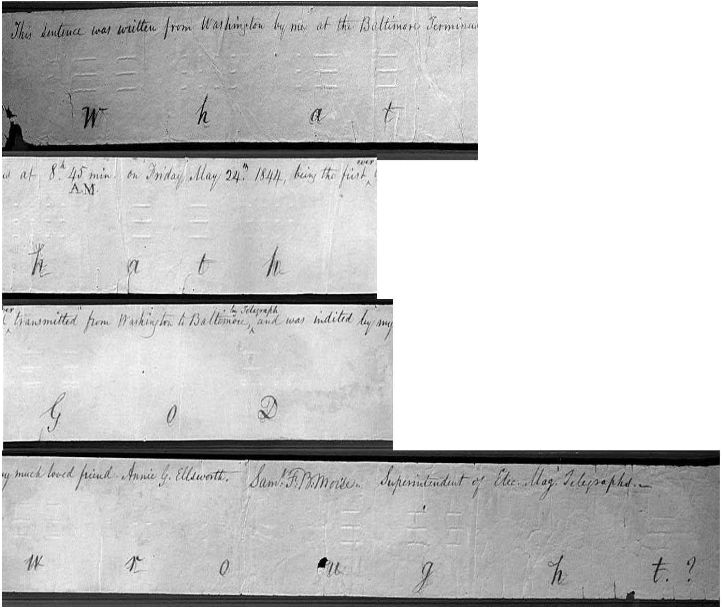
Степень сжатия

Пусть $A = \{a_1, \ldots, a_n\}$ – некоторый алфавит, $w \in A^+$ – текст, $\varphi \colon A \to X^+$ – кодирование с помощью X.

Обозначим через p_i относительную частоту появления знака $a_i \in A$ в тексте w, а через ℓ_i – длину слова $\varphi(a_i)$.
Тогда $L := \sum_{i=1}^{n} p_i \ell_i$ – средняя длина кодового слова (в расчете на один знак текста).
Именно этот параметр определяет степень сжатия:
чем он меньше, тем кодирование лучше.

Упражнение. Первой фразой, которую Морзе передал по телеграфу 24 мая 1844 года, была цитата из Книги Чисел: **WHAT HATH GOD WROUGHT** (Вот что творит Бог). Подсчитайте среднюю длину кодового слова при кодировании этой фразы кодом Морзе. Учтите, что код Морзе троичный: кодовый алфавит состоит из точки, тире и паузы. Скажем, код **E** – это точка и пауза. Код пробела между словами – две паузы.

Первая телеграмма



Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода.

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.

Применив неравенство Гиббса к $q_i := \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$, получим:

$$H \leqslant -\sum_{i=1}^{n} p_i \log \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$$

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.

Применив неравенство Гиббса к $q_i := \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$, получим:

$$H \leqslant -\sum_{i=1}^{n} p_i \log \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$$

по свойствам логарифмической функции

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.

Применив неравенство Гиббса к $q_i := \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$, получим:

$$H \leqslant -\sum_{i=1}^{n} p_i \log \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$$

по свойствам логарифмической функции

$$= \sum_{i=1}^{n} p_i \ell_i \log d + \sum_{i=1}^{n} p_i \log \sum_{i=1}^{n} d^{-\ell_i}$$

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.

Применив неравенство Гиббса к $q_i := \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$, получим:

$$H \leqslant -\sum_{i=1}^{n} p_i \log \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$$

по свойствам логарифмической функции

$$= \sum_{i=1}^{n} p_i \ell_i \log d + \sum_{i=1}^{n} p_i \log \sum_{i=1}^{n} d^{-\ell_i}$$

вторая сумма $\leqslant 0$ по неравенству Крафта–Макмиллана

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$. Применив неравенство Гиббса к $q_i := \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$, получим:

$$H \leqslant -\sum_{i=1}^{n} p_i \log \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$$

по свойствам логарифмической функции

$$= \sum_{i=1}^{n} p_i \ell_i \log d + \sum_{i=1}^{n} p_i \log \sum_{i=1}^{n} d^{-\ell_i}$$

вторая сумма $\leqslant 0$ по неравенству Крафта–Макмиллана

$$\leqslant \sum_{i=1}^{n} p_i \ell_i \log d$$

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$.

Применив неравенство Гиббса к $q_i := \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$, получим:

$$H \leqslant -\sum_{i=1}^{n} p_i \log \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$$

по свойствам логарифмической функции

$$= \sum_{i=1}^{n} p_i \ell_i \log d + \sum_{i=1}^{n} p_i \log \sum_{i=1}^{n} d^{-\ell_i}$$

вторая сумма $\leqslant 0$ по неравенству Крафта–Макмиллана

$$\leqslant \sum_{i=1}^{n} p_i \ell_i \log d = L \log d.$$

Нижняя оценка степени сжатия

Теорема Макмиллана позволяет оценить снизу среднюю длину кодового слова для любого кода. Нам потребуется вспомнить неравенство Гиббса $H := -\sum_{i=1}^{n} p_i \log p_i \leqslant -\sum_{i=1}^{n} p_i \log q_i$, где $0 \leqslant p_1, \ldots, p_n, q_1, \ldots, q_n \leqslant 1$, $\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1$.

Пусть $|X| = d$, а $C \subset X^+$ – код из n слов с длинами $\ell_1, \ldots, \ell_n$. Применив неравенство Гиббса к $q_i := \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$, получим:

$$H \leqslant -\sum_{i=1}^{n} p_i \log \frac{d^{-\ell_i}}{\sum_{i=1}^{n} d^{-\ell_i}}$$

по свойствам логарифмической функции

$$= \sum_{i=1}^{n} p_i \ell_i \log d + \sum_{i=1}^{n} p_i \log \sum_{i=1}^{n} d^{-\ell_i}$$

вторая сумма $\leqslant 0$ по неравенству Крафта–Макмиллана

$$\leqslant \sum_{i=1}^{n} p_i \ell_i \log d = L \log d. \qquad \text{Итак, } L \geqslant \frac{H}{\log d}.$$

Нижняя оценка степени сжатия, окончание

Вот, оказывается, в чем «информатический» смысл энтропии: отношение энтропии текста к логарифму размера кодового алфавита дает нижнюю оценку для ожидаемой длины кодового слова при любом кодировании текста (теорема Шеннона).

Нижняя оценка степени сжатия, окончание

Вот, оказывается, в чем «информатический» смысл энтропии: отношение энтропии текста к логарифму размера кодового алфавита дает нижнюю оценку для ожидаемой длины кодового слова при любом кодировании текста (теорема Шеннона).

Продолжение упражнения. Подсчитайте энтропию текста первой телеграммы **WHAT HATH GOD WROUGHT**.
Было ли кодирование этого текста кодом Морзе оптимальным?

Нижняя оценка степени сжатия, окончание

Вот, оказывается, в чем «информатический» смысл энтропии: отношение энтропии текста к логарифму размера кодового алфавита дает нижнюю оценку для ожидаемой длины кодового слова при любом кодировании текста (теорема Шеннона).

Продолжение упражнения. Подсчитайте энтропию текста первой телеграммы **WHAT HATH GOD WROUGHT**.
Было ли кодирование этого текста кодом Морзе оптимальным?

Для двоичного кодирования нижняя оценка упрощается до

$$L \geqslant H.$$

Точность нижней оценки

Можно показать, что оценка $L \geqslant \frac{H}{\log d}$ неулучшаема.

Точность нижней оценки

Можно показать, что оценка $L \geqslant \frac{H}{\log d}$ неулучшаема.

А именно, для любого кодового алфавита X такого, что $d := |X| > 1$, и любого исходного алфавита A можно так кодировать тексты из A^+ с помощью X, что средняя длина кодового слова в расчете на один знак из A будет сколь угодно близка к $\frac{H}{\log d}$.

Точность нижней оценки

Можно показать, что оценка $L \geqslant \frac{H}{\log d}$ неулучшаема.

А именно, для любого кодового алфавита X такого, что $d := |X| > 1$, и любого исходного алфавита A можно так кодировать тексты из A^+ с помощью X, что средняя длина кодового слова в расчете на один знак из A будет сколь угодно близка к $\frac{H}{\log d}$. (Правда, придется кодировать не каждый знак по отдельности, а блоки знаков.)

Точность нижней оценки

Можно показать, что оценка $L \geqslant \frac{H}{\log d}$ неулучшаема.

А именно, для любого кодового алфавита X такого, что $d := |X| > 1$, и любого исходного алфавита A можно так кодировать тексты из A^+ с помощью X, что средняя длина кодового слова в расчете на один знак из A будет сколь угодно близка к $\frac{H}{\log d}$. (Правда, придется кодировать не каждый знак по отдельности, а блоки знаков.)

Пусть p_i, $i = 1, \ldots, n$, – относительная частота i-го знака из A.

Точность нижней оценки

Можно показать, что оценка $L \geqslant \frac{H}{\log d}$ неулучшаема.

А именно, для любого кодового алфавита X такого, что $d := |X| > 1$, и любого исходного алфавита A можно так кодировать тексты из A^+ с помощью X, что средняя длина кодового слова в расчете на один знак из A будет сколь угодно близка к $\frac{H}{\log d}$. (Правда, придется кодировать не каждый знак по отдельности, а блоки знаков.)

Пусть p_i, $i = 1, \ldots, n$, – относительная частота i-го знака из A. Если ℓ_i – наименьшее натуральное число, удовлетворяющее неравенству $-\frac{\log p_i}{\log d} \leqslant \ell_i$, то $d^{-\ell_i} \leqslant p_i$.

Точность нижней оценки

Можно показать, что оценка $L \geqslant \frac{H}{\log d}$ неулучшаема.

А именно, для любого кодового алфавита X такого, что $d := |X| > 1$, и любого исходного алфавита A можно так кодировать тексты из A^+ с помощью X, что средняя длина кодового слова в расчете на один знак из A будет сколь угодно близка к $\frac{H}{\log d}$. (Правда, придется кодировать не каждый знак по отдельности, а блоки знаков.)

Пусть p_i, $i = 1, \ldots, n$, – относительная частота i-го знака из A. Если ℓ_i – наименьшее натуральное число, удовлетворяющее неравенству $-\frac{\log p_i}{\log d} \leqslant \ell_i$, то $d^{-\ell_i} \leqslant p_i$. Суммируя, получаем

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant \sum_{i=1}^{n} p_i = 1.$$

Точность нижней оценки

Можно показать, что оценка $L \geqslant \frac{H}{\log d}$ неулучшаема.

А именно, для любого кодового алфавита X такого, что $d := |X| > 1$, и любого исходного алфавита A можно так кодировать тексты из A^+ с помощью X, что средняя длина кодового слова в расчете на один знак из A будет сколь угодно близка к $\frac{H}{\log d}$. (Правда, придется кодировать не каждый знак по отдельности, а блоки знаков.)

Пусть p_i, $i = 1, \ldots, n$, – относительная частота i-го знака из A. Если ℓ_i – наименьшее натуральное число, удовлетворяющее неравенству $-\frac{\log p_i}{\log d} \leqslant \ell_i$, то $d^{-\ell_i} \leqslant p_i$. Суммируя, получаем

$$\sum_{i=1}^{n} d^{-\ell_i} \leqslant \sum_{i=1}^{n} p_i = 1.$$

По теореме Крафта есть префиксный код из n слов над X с длинами $\ell_1, \ldots, \ell_n$.

Точность нижней оценки, продолжение

Заметим, что по выбору чисел ℓ_i верны двойные неравенства

$$-\frac{\log p_i}{\log d} \leqslant \ell_i < -\frac{\log p_i}{\log d} + 1.$$

Точность нижней оценки, продолжение

Заметим, что по выбору чисел ℓ_i верны двойные неравенства

$$-\frac{\log p_i}{\log d} \leqslant \ell_i < -\frac{\log p_i}{\log d} + 1.$$

Умножая каждое из них на p_i и складывая, получаем

$$\frac{H}{\log d} \leqslant L < \frac{H}{\log d} + \sum_{i=1}^{n} p_i = \frac{H}{\log d} + 1.$$

Точность нижней оценки, продолжение

Заметим, что по выбору чисел ℓ_i верны двойные неравенства

$$-\frac{\log p_i}{\log d} \leqslant \ell_i < -\frac{\log p_i}{\log d} + 1.$$

Умножая каждое из них на p_i и складывая, получаем

$$\frac{H}{\log d} \leqslant L < \frac{H}{\log d} + \sum_{i=1}^{n} p_i = \frac{H}{\log d} + 1.$$

Возьмем произвольное натуральное m и применим описанное построение к алфавиту A^m, знаки которого – m-ки знаков из A.

Точность нижней оценки, продолжение

Заметим, что по выбору чисел ℓ_i верны двойные неравенства

$$-\frac{\log p_i}{\log d} \leqslant \ell_i < -\frac{\log p_i}{\log d} + 1.$$

Умножая каждое из них на p_i и складывая, получаем

$$\frac{H}{\log d} \leqslant L < \frac{H}{\log d} + \sum_{i=1}^{n} p_i = \frac{H}{\log d} + 1.$$

Возьмем произвольное натуральное m и применим описанное построение к алфавиту A^m, знаки которого – m-ки знаков из A. Вспомним: $H(Y, Z) = H(Y) + H(Z)$ при независимых Y и Z.

Точность нижней оценки, продолжение

Заметим, что по выбору чисел ℓ_i верны двойные неравенства

$$-\frac{\log p_i}{\log d} \leqslant \ell_i < -\frac{\log p_i}{\log d} + 1.$$

Умножая каждое из них на p_i и складывая, получаем

$$\frac{H}{\log d} \leqslant L < \frac{H}{\log d} + \sum_{i=1}^{n} p_i = \frac{H}{\log d} + 1.$$

Возьмем произвольное натуральное m и применим описанное построение к алфавиту A^m, знаки которого – m-ки знаков из A.
Вспомним: $H(Y, Z) = H(Y) + H(Z)$ при независимых Y и Z.
Поэтому энтропия A^m равна mH.

Точность нижней оценки, продолжение

Заметим, что по выбору чисел ℓ_i верны двойные неравенства

$$-\frac{\log p_i}{\log d} \leqslant \ell_i < -\frac{\log p_i}{\log d} + 1.$$

Умножая каждое из них на p_i и складывая, получаем

$$\frac{H}{\log d} \leqslant L < \frac{H}{\log d} + \sum_{i=1}^{n} p_i = \frac{H}{\log d} + 1.$$

Возьмем произвольное натуральное m и применим описанное построение к алфавиту A^m, знаки которого – m-ки знаков из A. Вспомним: $H(Y,Z) = H(Y) + H(Z)$ при независимых Y и Z. Поэтому энтропия A^m равна mH. Отсюда

$$\frac{mH}{\log d} \leqslant L_m < \frac{mH}{\log d} + 1.$$

Индекс m у средней длины появился потому, что при каждом m возникает свой код со своей средней длиной кодовых слов.

Точность нижней оценки, окончание

Итак,
$$\frac{mH}{\log d} \leqslant L_m < \frac{mH}{\log d} + 1.$$

Точность нижней оценки, окончание

Итак,

$$\frac{mH}{\log d} \leqslant L_m < \frac{mH}{\log d} + 1.$$

Деля почленно на m, получаем

$$\frac{H}{\log d} \leqslant \frac{L_m}{m} < \frac{H}{\log d} + \frac{1}{m}.$$

Точность нижней оценки, окончание

Итак,

$$\frac{mH}{\log d} \leqslant L_m < \frac{mH}{\log d} + 1.$$

Деля почленно на m, получаем

$$\frac{H}{\log d} \leqslant \frac{L_m}{m} < \frac{H}{\log d} + \frac{1}{m}.$$

По теореме о двух милиционерах $\frac{L_m}{m} \to \frac{H}{\log d}$ при $m \to \infty$.

Точность нижней оценки, окончание

Итак,

$$\frac{mH}{\log d} \leqslant L_m < \frac{mH}{\log d} + 1.$$

Деля почленно на m, получаем

$$\frac{H}{\log d} \leqslant \frac{L_m}{m} < \frac{H}{\log d} + \frac{1}{m}.$$

По теореме о двух милиционерах $\frac{L_m}{m} \to \frac{H}{\log d}$ при $m \to \infty$.

Но $\frac{L_m}{m}$ есть не что иное, как средняя длина кодового слова в расчете на один знак из A.

Точность нижней оценки, окончание

Итак,

$$\frac{mH}{\log d} \leqslant L_m < \frac{mH}{\log d} + 1.$$

Деля почленно на m, получаем

$$\frac{H}{\log d} \leqslant \frac{L_m}{m} < \frac{H}{\log d} + \frac{1}{m}.$$

По теореме о двух милиционерах $\frac{L_m}{m} \to \frac{H}{\log d}$ при $m \to \infty$.

Но $\frac{L_m}{m}$ есть не что иное, как средняя длина кодового слова в расчете на один знак из A. Ведь мы кодируем m-ки знаков и L_m – это средняя длина кодового слова для m-ки.

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании.

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании. Если энтропия текста равна H, любая запись этого текста, по которой он однозначно восстанови́м, требует не меньше H бит.

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании. Если энтропия текста равна H, любая запись этого текста, по которой он однозначно восстанови́м, требует не меньше H бит.

$\Rightarrow$ энтропия – полезная штука!

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании. Если энтропия текста равна H, любая запись этого текста, по которой он однозначно восстанови́м, требует не меньше H бит.

$\Rightarrow$ энтропия – полезная штука!

2. С точки зрения степени сжатия всего, чего можно добиться с помощью какого угодно кодирования, можно добиться и с помощью префиксного кодирования.

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании. Если энтропия текста равна H, любая запись этого текста, по которой он однозначно восстановúм, требует не меньше H бит.

$\Rightarrow$ энтропия – полезная штука!

2. С точки зрения степени сжатия всего, чего можно добиться с помощью какого угодно кодирования, можно добиться и с помощью префиксного кодирования.

$\Rightarrow$ префиксные коды рулят!

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании. Если энтропия текста равна H, любая запись этого текста, по которой он однозначно восстанови́м, требует не меньше H бит.

$\Rightarrow$ энтропия – полезная штука!

2. С точки зрения степени сжатия всего, чего можно добиться с помощью какого угодно кодирования, можно добиться и с помощью префиксного кодирования.

$\Rightarrow$ префиксные коды рулят!

3. Кодирование m-ок позволяет подойти сколь угодно близко к теоретической нижней оценке.

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании. Если энтропия текста равна H, любая запись этого текста, по которой он однозначно восстанови́м, требует не меньше H бит.

$\Rightarrow$ энтропия – полезная штука!

2. С точки зрения степени сжатия всего, чего можно добиться с помощью какого угодно кодирования, можно добиться и с помощью префиксного кодирования.

$\Rightarrow$ префиксные коды рулят!

3. Кодирование m-ок позволяет подойти сколь угодно близко к теоретической нижней оценке. На практике (при кодировании текстов на естественных языках) кодирование m-ок (m-грамм) существенно улучшает сжатие, ибо энтропия A^m, как правило, заметно меньше mH.

Выводы (langer Rede kurzer Sinn)

1. Энтропия дает по существу точную нижнюю оценку для средней длины кодового слова при произвольном кодировании. Если энтропия текста равна H, любая запись этого текста, по которой он однозначно восстанови́м, требует не меньше H бит.

$\Rightarrow$ энтропия – полезная штука!

2. С точки зрения степени сжатия всего, чего можно добиться с помощью какого угодно кодирования, можно добиться и с помощью префиксного кодирования.

$\Rightarrow$ префиксные коды рулят!

3. Кодирование m-ок позволяет подойти сколь угодно близко к теоретической нижней оценке. На практике (при кодировании текстов на естественных языках) кодирование m-ок (m-грамм) существенно улучшает сжатие, ибо энтропия A^m, как правило, заметно меньше mH.

$\Rightarrow$ m-граммы рулят!