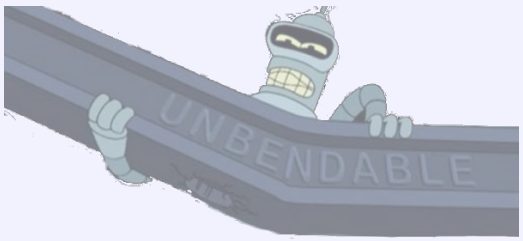
Коммуникационная сложность



Колмогоровская сложность

Теория информации

Конспект лекций

Теория кодирования

Содержание

Вместо введения

Данный текст представляет собой конспект курса по «Теории информации», прочитанный Дмитрием Соколовым в 2020 для второго курса ФМКН СПбГУ. Конспект составили Сергей Лучинин, Михаил Опанасенко; правил конспект Дмитрий Соколов.

1. Информация по Хартли

1.1. Базовые свойства

Пусть дано некоторое множество объектов. Мы хотим ввести некоторую *меру информации*, то есть хотим понять, сколько информации мы узнаём, получая некоторый элемент данного множества. Одна из общепринятых мер информации — количество бит. Попробуем формализовать эту меру информации

Определение 1.1 [Информация по Хартли]

Пусть A — некоторое конечное множество. За **информацию в множестве** A — будем принимать следующую величину:

$$\chi(A) := \log |A|.$$

Данное определение хорошо ложиться на интуитивное представление о том, что необходимо $\log |A|$ бит для описания некоторого элемента множества. Отметим, что число $\chi(A)$ может быть нецелым.

Замечание 1.2

В отличие от курса анализа, под log мы везде понимаем логарифм по основанию 2.

Попробуем описать свойства этого определения.

Утверждение 1.3

Пусть $A \subseteq X \times Y$ — конечное двумерное множество, A_X — его проекция на X, A_Y — на Y. Тогда выполнены следующие свойства:

1. $\chi(A) \geq 0$;
2. $\chi(A_X) \leq \chi(A), \chi(A_Y) \leq \chi(A)$;
3. $\chi(A) \leq \chi(A_X) + \chi(A_Y)$.

Доказательство. Следует из определения. □

Пользуясь этим определением мы можем доказать нетривиальные свойства множеств, например соотношение между «объёмом» и площадями проекций.

Упражнение 1.4

Для множества $A \subseteq \mathbb{N}^3$ будем обозначать $\pi_{ij}(A)$ проекцию A на координатную плоскость, задаваемую осями i, j (индексы $i, j \in [3]$). Докажите, что для любого конечного A выполняется:

$$2 \log |A| \leq \log |\pi_{12}(A)| + \log |\pi_{13}(A)| + \log |\pi_{23}(A)|.$$

Можем ли мы понять как изменится информация во множестве A, если мы уже про него что-то знаем? Аналогично определению 1.1 мы можем описать «условную информацию»,

содержащуюся в множестве A.

Определение 1.5

Пусть A — двумерное множество с проекциями X и Y. **Условную информация** содержащуюся в множестве A, если мы уже знаем вторую координату, определим следующим образом:

$$\chi_{Y|X}(A) := \max_x \big(\log |A_x|\big),$$

где A_x — сечение A по координате x.

Если говорить интуитивно, то эта мера нам описывает достаточное количество бит, нужное для кодирования элемента, зная его первую проекцию. Существенный недостаток это определения в том, что разным элементам могут соответствовать сечения разных размеров, а мы этого никак не учитываем.

Нетрудно проверить, что при таком определении выполнено неравенство

$$\chi(A) \le \chi(A_Y) + \chi_{X|Y}(A).$$

В дальнейшем иногда мы будем обозначать множество $\{1, 2, \dots, n\}$ через $[n]$.

1.2. Угадывание монетки

Симметричный вариант. Рассмотрим некоторые применения информации по Хартли. Пусть есть два игрока, первый загадывает число от 1 до n. Сколько вопросов с ответом «да/нет» необходимо задать второму игроку, чтобы угадать число? При этом у задачи есть два варианта: с *неадаптивной* стратегией, когда второй игрок пишет все вопросы заданы заранее, и *адаптивной* стратегией, когда второй игрок задаёт очередной вопрос, зная ответы на все предыдущие.

Для верхней оценки, как в адаптивной, так и не в адаптивной стратегии мы можем предъявить простую стратегию. Второй игрок может спросить каждый бит числа n в двоичной записи; поэтому количество запросов не превосходит $h = \lceil \log n \rceil$. Теперь давайте попробуем доказать, что ничего лучше сделать мы все равно не сможем.

Пусть Q_i — ответ на i-ый вопрос (один бит), N — искомое число,

$$B := Q_1 \times Q_2 \times \cdots \times Q_h.$$

Посмотрим на множество пар (N, B) по всем возможным N и B. Корректность протокола означает, что если мы знаем все Q_i, то можем определить число, то есть $\chi_B([n]) = 0$. Легко заметить, что $\chi(Q_i) \le 1$. Тогда:

$$\log n \le \chi(N, B) \le \sum_{i=1}^{h} \chi(Q_i) + \chi_B([n]) = \sum_{i=1}^{h} \chi(Q_i) \le h.$$

Таким образом, $h \ge \log n$, доказана нижняя оценка.

Ту же оценку можно было легко получить и другими, более простыми способами, но метод выше обобщается на гораздо более сложные ситуации.

Асимметричный вариант. Немного усложним задачу. Пусть за каждый ответ «да» второй игрок платит 1 монету, а за каждый ответ «нет» — 2 монеты.

Давайте попробуем адаптировать нашу стратегию для этого случая. Попробуем запросом делить множество «пополам» с точки зрения стоимости, то есть таким образом, чтобы при

ответе «нет» мы бы узнавали в два раза больше информации. (что, например, на первом шаге нам даст следующее соотношение: $2\chi_{Q_i=1}([n]) = \chi_{Q_i=0}([n])$).

Попробуем понять сколько нам потребуется заплатить при такой стратегии. Пусть Q_i — ответ на вопрос «верно ли, что загаданное число N лежит в множестве $T_i \subseteq X_i$?». Пусть X_i — множество элементов, в котором может лежать N после первых i вопросов. Наша стратегия говорит, что:

$$2(\chi(X_i) - \chi(T_i)) = \chi(X_i) - \chi(X_i \setminus T_i)$$

Распишем это по определению:

$$\begin{aligned} 2(\log|X_i| - \log|T_i|) = \log|X_i| - \log|X_i \setminus T_i| &\iff \\ \log|X_i| = 2\log|T_i| - \log|X_i \setminus T_i| &\iff \\ |X_i| = \frac{|T_i|^2}{|X_i \setminus T_i|}&. \end{aligned}$$

Обозначая $|X_i| = k$, $|T_i| = t$, получаем:

$$\begin{gathered} k = t^2/(k-t) \iff t^2 = k(k-t) = k^2 - kt \iff \\ t^2 + kt - k^2 = 0 \iff t = \frac{-k \pm \sqrt{k^2 + 4k^2}}{2} = k\left(\frac{-1 \pm \sqrt{5}}{2}\right). \end{gathered}$$

Таким образом, для реализации нашей стратегии на каждом шаге нужно выбирать такое T_i, что $\varphi|T_i| = |X_i|$, где φ — золотое сечение. Соответственно «средняя цена» бита информации будет $2(\chi(X_i) - \chi(T_i)) = 2\log\frac{1}{\varphi}$.

Поймем, что данная стратегия оптимальна. Не умаляя общности:

$$2(\chi(X_i) - \chi(T_i)) > \chi(X_i) - \chi(X_i \setminus T_i),$$

но в таком случае первый игрок может загадать такое число x, что $x \in T_i$, и второй игрок на этом шаге заплатим большую, чем «средняя», цену за бит информации.

Замечание 1.6

Конечно мы не можем поделить множество в иррациональной пропорции, но для больших n мы можем сколь угодно близко приблизится к этому.

Подобные игры с монетками используются в реальной жизни. В частности, размеры деревьев решений позволяют доказывать нижние оценки на различные алгоритмы для задачи выполнимости булевых формул. А для оценок на размеры деревьев решений используются игры с монетками. Рассмотрим пример.

Пример 1.7

Подобная стратегия применяется и в некоторых более современных задачах. Пусть есть $n+1$ голубь и n клеток. По принципу Дирихле нельзя посадить голубей в клетки таким образом, чтобы каждый сидел в клетке, и в одной клетке было бы не более одного голубя. Введем для каждой пары (голубь, клетка) переменную x_{ij}, будем считать, что $x_{ij} = 1$ означает, что i-ый голубь сидит в j-ой клетке, и $x_{ij} = 0$, если это не так. Тогда эти условия принципа Дирихле можно записать в виде невыполнимой системы уравнений:

1. для всех $i \in [n+1]$: $\prod_{j=1}^{n} (1 - x_{ij}) = 0$;
2. для всех i, i', j, где $i \neq i'$: $x_{ij} \cdot x_{i'j} = 0$.

Один игрок загадывает рассадку голубей, а второй пытается найти, какое из условий нарушено. В статье [BGL10] приведено «простое» доказательство нижней оценки на размер дерева решений для данной задачи, доказательство использует игру с монетками.

1.3. Взвешивание монеток

Рассмотрим еще один пример применения. Пусть даны n монеток, из которых одна фальшивая и имеет другой вес, и рычажные весы. Вопрос — можно ли за m взвешиваний определить фальшивую монету? Решите задачу в следующих вариантах:

1. $n = 30, m = 3$;
2. $n = 15, m = 3$;
3. $n = 14, m = 3$.

В отличие от предыдущей задачи, каждое взвешивание приносит больше информации: $\chi(Q_i) \leq \log_2 3$, так как возможны 3 ответа на каждый вопрос. Рассмотрим все варианты данной задачи.

1. При правильном протоколе должно быть выполнено неравенство
$$\log_2(30) = \chi([30]) \leq \sum_{i=1}^{3} \chi(Q_i) + \chi_B([30]) = \chi(Q_1) + \chi(Q_2) + \chi(Q_3) \leq \log_2(27),$$
что неверно. Значит, ответ — «нет».

2. В случае $n = 15$ оценка выше не даёт требуемого результата. Если добавить также условие, что надо определить, какая монета тяжелее, то надо рассматривать множество $[15] \times \{0, 1\}$, где 0 означает, что монета фальшива; и тогда верхняя оценка сработает.

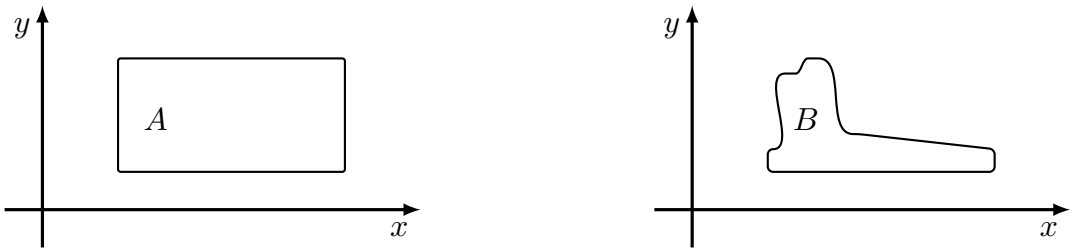
   Пусть надо только определить фальшивую монету. Заметим, что если хотя бы при одном взвешивании не было достигнуто равновесие, то мы можем определить не только фальшивую монету, но и то, тяжелее она или легче обычных. Пусть монетка, получающаяся как ответ при трёх равновесиях, имеет номер k. Тогда реально мы определяем информацию множества
   $$([15] \setminus \{k\}) \times \{0, 1\} \cup \{k\}$$
   порядка 29. Поскольку $29 > 27$, ответ по-прежнему нет.

3. Поскольку $2 \cdot 13 + 1 = 27$, то предыдущее рассуждение не работает. Однако ответ всё ещё «нет», но для доказательства нам понадобится некоторая теория.

2. Информация по Шеннону

2.1. Определение и свойства

В прошлом разделе мы увидели ряд проблем, возникающих при работе с информацией по Хартли. С одной стороны, у нас есть задача про 27 монет и 3 взвешивания, которую мы не понимаем как решать. С другой — определение «условной информации» ($\chi_{Y|X}(A)$) плохо описывает наше множество. Например, для следующих множеств выполнено равенство $\chi_{y|x}(A) = \chi_{y|x}(B)$, хотя сами множества ничем не похожи друг на друга (даже с точки знания количества объектов в них).



(a) (b)

Попробуем обобщить понятие информации для решения данных проблем. Введём новую меру информации μ, согласованную с определением по Хартли. Раньше мы предполагали, что все элементы в множестве A одинаковы. Теперь предположим, что каждый элемент появляется с некоторой вероятностью p_n; то есть μ будет задаваться уже не на множестве, а на распределении. В этих терминах свойство согласованности можно выразить следующим образом:

1. $\mu(U_n) = \log n$, где U_n — равномерное распределение n объектов (это и есть согласованность с предыдущим определением);
2. $\mu(p) \geq 0$, где p — любое распределение;
3. $\mu(p, q) = \mu(p) + \mu(q)$, где p и q — независимые распределения.

Мы можем дополнить этот набор аксиом свойством «непрерывности», а также утверждением «согласованности» с определением условной вероятности. Тогда набор аксиом можно переписать в следующем виде:

1. *монотонность*: если M, M' — равномерные распределения на $m \geq m'$ объектах соответственно, то $\mu(M) \geq \mu(M')$.
2. *аддитивность*: $\mu(p, q) = \mu(p) + \mu(q)$, где p и q — независимые распределения;
3. *непрерывность*: мера $\mu(B_p)$ непрерывна по p, где B_p — распределение нечестной монетки, которая выпадает решкой с вероятностью p, и орлом с вероятностью $1 - p$;
4. *согласованность с условной вероятностью*:

$$\mu(B, X) = \mu(B) + \Pr[B = 0] \cdot \mu(X \mid B = 0) + \Pr[B = 1] \cdot \mu(X \mid B = 1),$$

где B — распределение нечестной монетки, X — произвольное распределение и $\mu(\cdot \mid \cdot)$ означает применение меры к условному распределению.

В таком случае можно доказать (мы этого делать не будем), что мера μ с точностью до мультипликативной константы определяется по формуле $\mu(X) := \sum p_i \log \frac{1}{p_i}$.

Определение 2.1

Для случайной величины α с вероятностями событий $(p_1, p_2, \dots)$ меру

$$\mathrm{H}(\alpha) := \sum p_i \log \frac{1}{p_i}.$$

мы будем называть **энтропия** и обозначать H (иногда h).

Рассмотрим простые примеры.

1. Равномерное распределение: вероятность выпадения каждого элемента равна $\frac{1}{n}$.

$$\mathrm{H}(U_n) = \sum_{k=1}^{n} \frac{1}{n} \log n = \log n.$$

2. Нечестная монетка:

$$\mathrm{H}(B_p) = p \log \frac{1}{p} + (1-p) \log \frac{1}{1-p}$$

— *бинарная энтропия*. Её часто обозначают через $\mathrm{H}(p)$.

Поскольку теорему Шеннона мы оставили без доказательства, то нужно проверить, что энтропия удовлетворяет нашим аксиомам.

Утверждение 2.2

Энтропия $\mathrm{H}(\alpha)$ обладает следующими свойствами:

1. $\mathrm{H}(\alpha) \leq \log |\alpha|$, где α — произвольное распределение и $|\alpha|$ — размер носителя;
2. $\mathrm{H}(\alpha, \beta) \leq \mathrm{H}(\alpha) + \mathrm{H}(\beta)$, где α, β — произвольные распределения.

Доказательство. Оба пункта мы будем доказывать похожим способом — при помощи неравенства Йенсена.

1. Распишем по определению и применим неравенство:

$$\mathrm{H}(\alpha) = \sum_{i=1}^{n} p_i \log \frac{1}{p_i} \leq \log \left(\sum_{i=1}^{n} p_i \frac{1}{p_i} \right) = \log n = \log |\alpha|.$$

2. Положим $p_{ij} = \Pr[\alpha = i, \beta = j]$, $p_{i\cdot} = \Pr[\alpha = i]$, $p_{*j} = \Pr[\beta = j]$. Заметим, что

$$p_{i\cdot} = \sum_j p_{ij}, \qquad p_{\cdot j} = \sum_i p_{ij},$$

— вероятность того, что выпал элемент i, равна вероятности того, что выпал элемент i и какой-то элемент j в β.

В этих терминах мы можем описать энтропию пары:

$$\mathrm{H}(\alpha, \beta) = \sum_{i,j} p_{ij} \cdot \log \frac{1}{p_{ij}},$$

а также выразить сумму энтропий:

$$\mathrm{H}(\alpha)+\mathrm{H}(\beta)=\sum_i p_{i\cdot}\cdot\log\frac{1}{p_{i\cdot}}+\sum_j p_{\cdot j}\log\frac{1}{p_{\cdot j}}=\sum_{ij}\left(p_{ij}\cdot\log\frac{1}{p_{i\cdot}}+p_{ij}\cdot\log\frac{1}{p_{\cdot j}}\right).$$

Тогда по неравенству Йенсена:

$$\mathrm{H}(\alpha,\beta)-\mathrm{H}(\alpha)-\mathrm{H}(\beta)=\sum_{ij}p_{ij}\log\frac{p_{i\cdot}p_{\cdot j}}{p_{ij}}\le\log\left(\sum_{i,j}p_{i\cdot}p_{\cdot j}\right)=\log 1=0.$$

Отметим, что если α и β независимы, то $p_{i\cdot}p_{\cdot j}=p_{ij}$, и мы получаем равенство $\mathrm{H}(\alpha,\beta)=\mathrm{H}(\alpha)+\mathrm{H}(\beta)$.

□

Теперь перейдем к определению условной энтропии.

Определение 2.3

Энтропией α при $\beta=b$ мы будем называть энтропию распределения α при условии, что $\beta=b$, то есть следующую величину:

$$\mathrm{H}(\alpha\mid\beta=b):=\sum_i\Pr[\alpha=i\mid\beta=b]\cdot\log\frac{1}{\Pr[\alpha=i\mid\beta=b]}$$

Тогда **энтропией α при условии β** мы назовем среднее значение по b энтропии α при $\beta=b$. Таким образом:

$$\mathrm{H}(\alpha\mid\beta):=\mathop{\mathbb{E}}_{b\sim\beta}[\mathrm{H}(\alpha\mid\beta=b)]=\sum_b\mathrm{H}(\alpha\mid\beta=b)\cdot\Pr[\beta=b].$$

Условная энтропия обладает естественными базовыми свойствами.

Утверждение 2.4

1. $\mathrm{H}(\alpha\mid\beta)\ge 0,\mathrm{H}(f(\alpha)\mid Y)=0.$
2. $\mathrm{H}(\alpha,\beta)=\mathrm{H}(\alpha)+\mathrm{H}(\beta\mid\alpha).$
3. $\mathrm{H}(\alpha)\ge\mathrm{H}(\alpha\mid\beta).$

Мы приведем доказательство третьего свойства, а первое и второе оставим в качестве упражнения. **Доказательство.** По неравенству Йенсена для логарифма:

$$\mathrm{H}(\alpha\mid\beta)-\mathrm{H}(\alpha)=\sum_{i,j}\left(p_{ij}\log\frac{1}{p_{i|j}}-p_{ij}\log\frac{1}{p_{i\cdot}}\right)=\sum_{i,j}p_{ij}\cdot\log\frac{p_{i\cdot}}{p_{i|j}}\le 0,$$

где $p_{i|j}=\Pr[\alpha=i\mid\beta=j]$. □

Данное свойство обобщается естественным образом на условную энтропию.

Упражнение 2.5

Докажите, что $\mathrm{H}(\alpha\mid\beta)\ge\mathrm{H}(\alpha\mid\beta,\gamma)$.

2.2. Применения энтропии

Еще немного о взвешивании. Вернемся к последней задаче из раздела 1.3. Теперь мы готовы решить про 14 монеток и 3 взвешивания.

Предположим, что в стратегии при трёх равенствах мы получаем монету с номером i. Как мы помним, нельзя определить, тяжелее она или легче других монет, в то время как для всех остальных монет это узнать можно. Это значит, что в дереве решения (в нём 27 листьев), i встречается среди листьев только один раз, а остальные индексы — два раза, причём один раз в ветке меньше, а в другой раз в ветке больше.

Зададим следующее распределение на монетках:

$$p_k = \begin{cases} \frac{1}{27}, & \text{если } k = i, \\ \frac{2}{27}, & \text{если } k \neq i, \end{cases}$$

и распределение на парах из монетки и больше/меньше: $p_{(k,<)} = p_{(k,>)} = p_k/2$.

Заметим, что если существует дерево решений с тремя взвешиваниями, то распределение p индуцирует равномерное распределение на листьях ℓ. С другой стороны мы знаем, что лист дерева определяется результатами трёх взвешиваний, назовём их q_1, q_2, q_3. Таким образом:

$$\begin{aligned} \mathrm{H}(\ell) &= 3 \log 3 \\ \mathrm{H}(\ell \mid q_1, q_2, q_3) &= 0. \end{aligned}$$

Скомбинируем все вместе:

$$3 \log 3 = \mathrm{H}(\ell) \leq \mathrm{H}(q_1, q_2, q_3) + \mathrm{H}(\ell \mid q_1, q_2, q_3) = \mathrm{H}(q_1, q_2, q_3) \leq \mathrm{H}(q_1) + \mathrm{H}(q_2) + \mathrm{H}(q_3).$$

Однако $\mathrm{H}(q_j) \leq \log 3$, так как мы выбираем из трёх вариантов, а энтропия не превосходит логарифма размера носителя. Таким образом, единственная возможность, когда неравенство выполнено, это когда все три исхода равновероятны. А это значит, что на первом шаге мы с равной вероятностью идём по трём разным веткам от вершины дерева.

Пусть на первом шаге взвешивается по k монет на каждой чаше. Вероятность того, что левая чаша перевесила, равна $\frac{2k}{27}$, так как либо фальшивая монета легче и лежит слева, либо она тяжелее и лежит справа; а вероятность того, что на левой чаше оказалась фальшивая монета легче настоящей равна вероятности того, что на правой чаше оказалась фальшивая монета тяжелее настоящей и равна $k/27$. Чтобы это число равнялось одной трети, нужно взять $k = 4.5$, что невозможно. Противоречие.

Оценка на биномиальные коэффициенты. Теперь попробуем получить оценку на биномиальные коэффициенты при помощи свойств энтропии.

Утверждение 2.6 [Оценка на биноминальные коэффициенты]

Для произвольного n и $k \leq n/2$ выполнено неравенство

$$C = \sum_{i=0}^{k} \binom{n}{i} \leq 2^{n\,\mathrm{H}\left(\frac{k}{n}\right)}.$$

Доказательство. Рассмотрим n объектов, из них выберем не более, чем k штук. Пусть X соответствует равномерному распределению по таким множествам. Как следствие $\mathrm{H}(X) = \log C$. При этом:

$$\mathrm{H}(X) \leq \mathrm{H}\left(X_1, X_2, X_3, \dots, X_i\right) \leq \sum \mathrm{H}(X_i),$$

где X_i — вероятность того, что мы выбрали i-ый элемент (проекция распределения X на i координату). По построению все эти распределения одинаковы, таким образом $\mathrm{H}(X) \leq n\,\mathrm{H}(X_1)$. Но заметим, что вероятность, с которой мы можем выбрать первый элемент, не больше, чем $\frac{k}{n}$, то есть $\mathrm{H}(X_1) = h\left(\frac{k}{n}\right)$ (так как мы берём распределение на множествах мощности не более k). Откуда следует искомое неравенство. □

«Треугольники» и «углы» в графах. Пусть дан ориентированный граф без кратных рёбер и петель. Упорядоченную тройку (x, y, z) вместе в рёбрами из x в y, из y в z, и из z в x, будем называть **треугольником**. **Углом** мы будем называть упорядоченную тройку вершин (x, y, z) вместе с рёбрами из x в y и из x в z. В частности, любое ребро (x, y) является углом, так как можно взять $z = y$.

Теорема 2.7 [[KR11]]

В любом графе число треугольников не превосходит числа углов.

Доказательство. Пусть X, Y, Z — случайные величины, соответствующие первой, второй и третьей вершине треугольника в равномерном распределении на треугольниках соответственно. Тогда $\mathrm{H}(X, Y, Z) = \log|\triangle|$. С другой стороны:

$$\mathrm{H}(X, Y, Z) = \mathrm{H}(X) + \mathrm{H}(Y, Z \mid X) = \mathrm{H}(X) + \mathrm{H}(Y \mid X) + \mathrm{H}(Z \mid X, Y).$$

Заметим, что если убрать X из $\mathrm{H}(Z \mid X, Y)$, то энтропия только возрастёт. Следовательно:

$$\mathrm{H}(X, Y, Z) \leq \mathrm{H}(X) + \mathrm{H}(Y \mid X) + \mathrm{H}(Z \mid Y).$$

Картинка симметрична (можно получить одно распределение из другого циклическим сдвигом), поэтому $\mathrm{H}(Y \mid X) = \mathrm{H}(Z \mid Y)$ и

$$\mathrm{H}(X, Y, Z) \leq \mathrm{H}(X) + 2\,\mathrm{H}(Y \mid X).$$

Определим распределение на углах. Выбираем вершину с той же вероятностью, с которой она является первой вершиной некоторого треугольника, обозначаем её через x. Выбираем равновероятно какой-то треугольник с вершиной x, проводим ребро и обозначаем вторую вершину y. Потом ещё раз независимо выбираем треугольник и проводим ребро в z. Посчитаем энтропию:

$$\mathrm{H}(X, Y, Z) = \mathrm{H}(X) + \mathrm{H}(Y \mid X) + \mathrm{H}(Z \mid X, Y).$$

Поскольку при известном X величины Y и Z независимы, то $\mathrm{H}(Z \mid X, Y) = \mathrm{H}(Z \mid X)$. Поскольку Y и Z выбираются одинаковым образом, $\mathrm{H}(Y \mid X) = \mathrm{H}(Z \mid X)$. Таким образом,

$$\mathrm{H}(X, Y, Z) = \mathrm{H}(X) + 2\,\mathrm{H}(Y \mid X).$$

Осталось заметить, что в обоих распределениях X выбирается одинаковым образом, и Y при известном X тоже выбирается также. Значит, энтропия некоторого распределения на углах не менее $\log|\triangle|$. Значит, углов не меньше, чем треугольников. □

3. Теория кодирования

Определение 3.1

Будем называть **кодом** функцию $C\colon \{a_1, \ldots, a_n\} \to \{0,1\}^*$, сопоставляющую буквам некоторого алфавита кодовые слова. Если любое сообщение, которое получено применением кода C, декодируется однозначно (то есть единственным образом разрезается на образы C), то такой код будем называть **однозначно декодируемым**.

3.1. Префиксные коды

Довольно удобно иметь более сильное свойство кода, чем однозначная декодируемость, которое позволяет декодировать сообщение отдельно по буквам.

Определение 3.2

Будем называть код **префиксным** (**беспрефиксным**, **prefix-free**), если никакое кодовое слово не является префиксом другого кодового слова.

Давайте попробуем понять, что любой однозначно декодируемый код можно переделать в беспрефиксный. Для этого мы попробуем описать критерии существования кодов.

Теорема 3.3

Пусть набор целых чисел $\ell_1, \dots, \ell_n$ удовлетворяет неравенству

$$\sum_{i=1}^{n} 2^{-\ell_i} \leq 1,$$

тогда существует префиксный код с кодовыми словами $c_1, \dots, c_n$, где $|c_i| \leq \ell_i$.

Доказательство. Доказательство этого утверждения мы оставим в качестве упражнения. □

Если мы покажем, что для любого однозначно декодируемого кода следующее неравенство всегда выполнено, то вместе с теоремой 3.3 это позволит переделывать одни коды в другие.

Утверждение 3.4 [Неравенство Крафта–Макмиллана]

Для любого однозначно декодируемого кода с кодовыми словами $c_1, c_2, \dots, c_n$ выполнено неравенство

$$\sum_{i=1}^{n} 2^{-|c_i|} \leq 1.$$

Доказательство. Доказательство этой теоремы должно использовать однозначную декодируемость «в полном объёме», то есть для любой длины декодируемых сообщений. Формально заменим в каждом c_i нули на буквы x, а единицы на буквы y, где x и y не коммутируют. Пусть $p_i(x, y)$ — моном, соответствующий c_i (например, коду 010 соответствует моном xyx); L — большое натуральное число. Рассмотрим следующий полином:

$$P^L(x, y) = \left(\sum_i p_i(x, y) \right)^L \leq \sum_{i=L}^{L \cdot \max |c_i|} M_i(x, y),$$

где M_i — это сумма всевозможных мономов степени i. Неравенство выполнено, так как код однозначно декодируемый, а значит каждый моном в левой части есть и в правой.

Полагая $x = y = \frac{1}{2}$, получаем:

$$P^L\left(\frac{1}{2}, \frac{1}{2}\right) \leq \sum_{i=L}^{L \cdot \max |c_i|} (2^i \cdot 2^{-i}) \leq \mathcal{O}(L).$$

Теперь предположим, что неравенство Крафта–Макмиллана не выполнено для данного кода. Тогда

$$\sum_i p_i\left(\frac{1}{2}, \frac{1}{2}\right) = \sum 2^{-|c_i|} = 1 + \varepsilon > 1.$$

Значит, $P^L = (1+\varepsilon)^L > \mathcal{O}(L)$, что противоречит предыдущему рассуждению о линейности роста. □

Из предыдущих двух теорем следует, что по любому однозначно декодируемому коду можно построить префиксный код с теми же длинами кодов.

Теорема 3.5 [Шеннон]

Для любого распределения p и однозначно декодируемого кода выполнено неравенство

$$\sum_i p_i \, |c_i| \geq \mathrm{H}(p),$$

где p_i — вероятность, с которой встречается буква i, а c_i — её код.

Доказательство. Доказательство следует из неравенства Йенсена и неравенства Крафта–Макмиллана,

$$\mathrm{H}(p) - \sum_i p_i \, |c_i| = \sum_i p_i \cdot \log \frac{2^{-|c_i|}}{p_i} \leq \log \left(\sum_i p_i \cdot \frac{2^{-|c_i|}}{p_i} \right) \leq 0,$$

что и требовалось. □

Теорема 3.6

Для любого распределения p существует такой префиксный код, что

$$\sum_i p_i \, |c_i| \leq \mathrm{H}(p) + 1.$$

Доказательство. Пусть $|c_i| = \lceil \log \frac{1}{p_i} \rceil$. В таком случае неравенство из условия выполнено, так как $p_i|c_i| \leq p_i \log \frac{1}{p_i} + p_i$, и $\sum p_i = 1$. Кроме того:

$$\sum_i 2^{-|c_i|} = \sum_i 2^{-\lceil \log \frac{1}{p_i} \rceil} \leq \sum_i p_i = 1.$$

Для завершения доказательства заметим, что по теореме 3.3 существует префиксный код, удовлетворяющий этому неравенству. Этот код и будет удовлетворять условию теоремы. □

3.2. Примеры эффективных кодов

Код Шеннона–Фано. Отсортируем вероятности, $p_1 \geq p_2 \geq \ldots \geq p_n$. «Уложим» вероятности p_i в отрезок $[0, 1]$, получая таким образом точки:

$$0 \leq p_1 < p_1 + p_2 < \cdots < p_1 + p_2 + \cdots + p_n \leq 1.$$

Разобьём интервал пополам, и скажем, что все коды, отвечающие точкам слева от разреза, начинаются с нуля, а точкам справа — с единицы. Если отрезок пересекает разрез, и он самый левый (первый), то соответствующий код начинается с 0; если отрезок пересекает разрез и он самый правый (последний), то код начинается с 1. Иначе выбираем ноль или единицу произвольным образом. Продолжаем рекурсивно этот процесс, пока в интервале не останется ровно один отрезок.

Упражнение 3.7

Для кода Шеннона–Фано выполнено равенство:

$$\sum_{i=1}^{n} p_i \left|c_i\right| = \mathrm{H}(p) + \mathcal{O}\left(1\right), \qquad n \to \infty.$$

Код Хаффмана. Код Хаффмана строится индуктивно. При $n = 2$ кодовые слова — $c_1 = 0, c_2 = 1$. При $n > 2$ рассмотрим два символа a и b с минимальными вероятностями p_{n-1} и p_n. Заменим указанные символы на новый символ σ, и дадим ему вероятность $p := p_{n-1}+p_n$. Построим код Хаффмана для $n-1$ символа, и обозначим код символа σ за c, после чего скажем, что код символа a — это $c0$, а код символа b — $c1$.

Теорема 3.8 [Хаффман]

Для кода Хаффмана выполнено неравенство:

$$\sum_{i} p_i \left|c_i\right| \leq \mathrm{H}(p) + 1, \tag{1}$$

и для любого другого однозначно декодируемого кода c_i' выполнено неравенство

$$\sum p_i \left|c_i'\right| \geq \sum p_i \left|c_i\right|. \tag{2}$$

Доказательство. Докажем неравенство (2), тогда из него и теоремы 3.6 будет следовать неравенство (1).

Предположим, что есть некоторый префиксный код, для которого оно нарушено. Рассмотрим такой код с минимальным числом символов. Посмотрим на два символа с самым длинным кодом. Мы хотим сказать, что они имеют самые маленькие вероятности. Действительно, если бы это было не так, то коды символа с большей вероятностью и меньшей можно было бы поменять местами, при этом средняя длина кода от этого бы только уменьшилась. Можно считать, что коды этих двух символов равны $v0$ и $v1$ соответственно(упражнение). «Склеим» эти два символа, как в коде Хаффмана.

Получившийся код мы можем переделать в код Хаффмана так, чтобы средняя длина не увеличилась. Это можно сделать, так как мы брали код с минимальным числом символов, для которого нарушается неравенство. Осталось заметить, что если «расклеить» символы, то мы получим в точности код Хаффмана, так как в нём мы делали то же самое первое действие (склеивали вершины с минимальной вероятностью). Отсюда следует, что наше предположение неверно, а значит неравенство (2) выполнено всегда. □

Арифметическое кодирование. Назовём стандартным интервалом интервал вида $[0.v0, 0.v1)$, где v — некоторая последовательность битов. Уложим вероятности p_i в отрезок $[0, 1]$, получатся точки

$$0 \leq p_1 < p_1 + p_2 < \cdots < p_1 + p_2 + \cdots + p_n \leq 1.$$

Пусть $[0.v_i0, 0.v_i1)$ — максимальный стандартный интервал в отрезке

$$[p_1 + p_2 + \cdots + p_{i-1}, p_1 + p_2 + \cdots + p_i].$$

Тогда сопоставим i-ой букве код v_i0. Заметим, что код получился префиксным, так как если v_i является префиксом v_j, то интервал $[0.v_j0, 0.v_j1)$ вложен в интервал $[0.v_i0, 0.v_i1)$, а такого при построении v_i не может произойти.

Лемма 3.9

В отрезке $[a, b]$ длина наибольшего стандартного интервала не меньше, чем $\frac{b-a}{8}$.

Доказательство. Доказательство мы оставим в качестве упражнения. □

Утверждение 3.10

Для арифметического кода выполняется неравенство:

$$\sum_i p_i \, |v_i| \leq \mathrm{H}(p) + 2.$$

Доказательство. Из леммы 3.9 следует, что если $|v_i| = k$, то:

$$0.v_i 1 - 0.v_i 0 = 2^{-k-1} \geq \frac{p_i}{8}.$$

Отсюда следует, что $k + 1 \leq \log \frac{8}{p_i}$, а значит $|v_i| = k \leq \log \frac{1}{p_i} + 2$, и

$$\sum_i p_i \, |v_i| \leq \mathrm{H}(p) + 2(p_1 + p_2 + \cdots + p_n) \leq \mathrm{H}(p) + 2.$$

□

3.3. Кодирование с ошибками

Пусть $p_1, \ldots, p_k$ — вероятности, с которыми встречаются буквы в алфавите. Будем рассматривать слова фиксированной длины n, которые будут кодироваться в слова заданной длины L_n. Пусть нам даны функции кодирования и декодирования:

$$E\colon [k]^n \to \{0,1\}^{L_n}, \qquad D\colon \{0,1\}^{L_n} \to [k]^n.$$

При этом мы отказываемся от условия, что код декодируется однозначно, но требуем, чтобы вероятность $\varepsilon_n := \Pr[D(E(w)) \neq w]$ стремилась к нулю при $n \to \infty$.

Теорема 3.11 [Шеннон]

1. Если $L_n = \lceil h \cdot n \rceil$, где $h > \mathrm{H}(p)$, то существуют такие функции E, D, что $\varepsilon_n \to 0$.
2. Если $L_n = \lceil h \cdot n \rceil$, где $h < \mathrm{H}(p)$, то для любых E, D последовательность ε_n стремится к единице.

Доказательство. Пусть w — слово длины n. Будем говорить, что буква i является δ-**типичной**, если $|n_i/n - p_i| \leq \delta$ где n_i — количество букв i в w. Соответственно, w будем называть δ-**типичным**, если это неравенство выполняется для всех букв i. Зафиксируем $\delta := n^{-0.49}$ и рассмотрим случайную величину X_{ij} — характеристическую функцию того, что в позиции j находится буква i. Тогда для случайной величины: $X_i := \sum\limits_j X_{ij}$ мы можем написать неравенство Чебышёва:

$$\Pr[|X_i - \mu| \geq \delta n] \leq \frac{\mathrm{Var}[X_i]}{(\delta n)^2} = \frac{np_i(1-p_i)}{(\delta n)^2} = \mathcal{O}\left(n^{-0.02}\right),$$

где $\mu := \mathbb{E}[X_i] = np_i$. Таким образом, доля слов, в которых буква i нетипична стремится к нулю, а поскольку число букв фиксировано мы можем заключить, что и в целом доля нетипичных слов стремится к нулю.

Число слов с заданным количеством вхождений каждой буквы равно:

$$N_{n_1,n_2,\dots,n_k} = \frac{n!}{n_1!n_2!\dots n_k!}.$$

Применим оценку $n! = \mathsf{poly}(n) \cdot (n/e)^n$. Тогда:

$$\log N_{n_1,n_2,\dots,n_k} = \log\left(\left(\frac{n}{n_1}\right)^{n_1}\left(\frac{n}{n_2}\right)^{n_2}\dots\left(\frac{n}{n_k}\right)^{n_k}\right) + \mathcal{O}(\log n).$$

Оценим это выражение:

$$\log\left(\left(\frac{n}{n_1}\right)^{n_1}\left(\frac{n}{n_2}\right)^{n_2}\dots\left(\frac{n}{n_k}\right)^{n_k}\right) = \sum n(p_i+\delta_i)\log\frac{1}{p_i+\delta_i}$$
$$\leq n\,\mathrm{H}(p) + \mathcal{O}\left(\max_i \delta_i n\right).$$

Есло слово, типично, то $|\delta_i| = |n_i - np_i| \leq \delta n$ и следовательно в таком случае:

$$\log N_{n_1,n_2,\dots,n_k} \leq n\,\mathrm{H}(p) + \mathcal{O}(\delta n).$$

И таким образом количество типичных слов не превосходит:

$$\sum_{n_i \in [n(p_i-\delta), n(p_i+\delta)]} N_{n_1,n_2,\dots,n_k} \leq n^k 2^{n\,\mathrm{H}(p)+\mathcal{O}(\delta n)} < 2^{h\cdot n}.$$

Поскольку типичных слов мало, кодер может инъективно закодировать типичные слова и “проигнорировать” все остальные.

Теперь перейдем к доказательству второго случая: $h < \mathrm{H}(p)$.

Пусть ε'_n вероятность ошибки при декодировании δ-типичных слов. Нам достаточно показать, что $\varepsilon'_n \to 1$, поскольку $|\varepsilon'_n - \varepsilon|$ не превосходит вероятности того, что слово нетипично, т.е. не более $\mathcal{O}\left(n^{-0.02}\right)$.

Давайте рассмотрим конкретное δ-типичное слово w. Оценим вероятность появления w:

$$\Pr[w] = p_1^{n_1} \cdot \dots \cdot p_k^{n_k} = 2^{-\sum n_i \log \frac{1}{p_i}} \leq 2^{-\sum (p_i+\delta_i)\log\frac{1}{p_i}\cdot n}.$$

Поскольку декодер D — детерминированный алгоритм принимающий на вход строку $x \in \{0,1\}^{L_n}$, то область значения D имеет размер не более 2^{L_n}. Заметим, что $D(E(w)) = w$ означает, что w принадлежит области значений D. Таким образом:

$$\Pr_w[D(E(w)) = w] \leq$$
$$\Pr_w[w \in \mathrm{Im}(D)] \leq 2^{L_n} \max_w \Pr[w] \leq$$
$$2^{-\mathrm{H}(\alpha)\cdot n + \mathcal{O}(\delta\cdot n)} \leq$$
$$2^{h\cdot n - \mathrm{H}(\alpha)\cdot n + \mathcal{O}(\delta\cdot n)} \to 0.$$

□

4. Приложения теории информации к криптографии

О криптографических применениях нам будет удобно говорить в терминах «взаимной информации».

Определение 4.1

Взаимной информацией между случайными величинами α и β будем называть величину:

$$I(\alpha\colon\beta) \coloneqq \mathrm{H}(\alpha) - \mathrm{H}(\alpha \mid \beta).$$

Также определим **взаимную информацию в α и β при условии γ**:

$$I(\alpha\colon\beta \mid \gamma) \coloneqq \mathrm{H}(\alpha \mid \gamma) - \mathrm{H}(\alpha \mid \beta, \gamma).$$

Взаимная информация обладает естественными свойствами:

1. $I(\alpha\colon\beta) = I(\beta\colon\alpha)$;
2. α и β независимы тогда и только тогда, когда $I(\alpha\colon\beta) = 0$;
3. $I(f(\alpha)\colon\beta) \leq I(\alpha\colon\beta)$ для любой функции f.
4. $I(\alpha\colon\beta) = \mathrm{H}(\alpha, \beta) - \mathrm{H}(\alpha \mid \beta) - \mathrm{H}(\beta \mid \alpha)$.

Доказательство, которых мы оставим в качестве упражнения.

4.1. Шифрование с закрытым ключом

Рассмотрим схему передачи слова от Алисы к Бобу. Мы предполагаем, что Алисе и Бобу заранее известен некоторый ключ k, но этот ключ не известен злоумышленнику (Чарли).

Алиса получает на вход слово w. С помощью ключа k и алгоритма E она кодирует слово w и отправляет получившееся слово $c \coloneqq \mathrm{E}(w, k)$ Бобу. Боб декодирует полученное слово с помощью ключа k и алгоритма дешифровки D и получает первоначальное слово $\mathrm{D}(c, k) = w$. Может случиться так, что Чарли перехватил сообщение и получил слово c. Мы бы хотели, чтобы он не смог восстановить исходное слово w, зная c, то есть:

$$\begin{cases} \mathrm{H}(c \mid w, k) = 0, & c \text{ определяется значениями } w \text{ и } k \\ \mathrm{H}(w \mid c, k) = 0, & \text{зная ключ } k \text{ и сообщение } c \text{ Боб может восстановить слово } w \\ I(c\colon w) = 0. & \mathrm{H}(w) = \mathrm{H}(w \mid c) \end{cases}$$

Последнее условие нам говорит, что даже при том, что Чарли знает c, для восстановления w ему необходимо столько же битов информации, как и без знания c. То есть знание c никак не помогло Чарли. Если выполнены все три условия, то назовём эту схему **идеальной**.

Замечание 4.2

Первое условие можно опустить, что соответствует тому, что алгоритм шифрования E будем вероятностным.

Рассмотрим пример идеальной схемы. Пусть $\mathrm{E}(w, k) = w \oplus k$, где слово w выбирается из множества слов длины n, k — это ключ, известный заранее, $|k| = n$. Заметим, что для этого кодирования выполнены первые 2 условия. Также несложно проверить третье условие.

В этой схеме ключевой момент, что нам потребовался ключ такой же длины, как и длина сообщения. Возникает естественный вопрос можно ли сделать что-то более эффективное.

Теорема 4.3 [Шеннон]

Для идеальной схемы шифрования с закрытым ключом выполнено $\mathrm{H}(k) \geq \mathrm{H}(w)$, даже в случае вероятностного алгоритма шифрования.

Доказательство. Доказательство следует из неравенств:

$$\mathrm{H}(w) = \mathrm{H}(w \mid c) \leq \mathrm{H}(w, k \mid c) = \mathrm{H}(k \mid c) + \mathrm{H}(w \mid c, k) = \mathrm{H}(k \mid c) \leq \mathrm{H}(k).$$

□

4.2. Схема разделения секрета

Пусть у нас есть некоторый секрет S_0 и n участников и мы хотим раздать игрокам «ключи» $S_1, \ldots, S_n$ таким образом, чтобы они могли узнать секрет S_0 только все вместе, а любое подмножество участников — не могло.

Попробуем переформулировать нашу задачу более формально. Пусть $(S_0, S_1, \ldots, S_n)$ — это набор случайных величин. Будем называть его **схемой разделения секрета**, если:

1. $\mathrm{H}(S_0 \mid S_1, \ldots, S_n) = 0$;
2. $\mathrm{H}(S_0 \mid S_I) = \mathrm{H}(S_0)$, где $I \subsetneq [n]$.

Пример 4.4

Пусть $S_0 \in \{0,1\}^\ell$, $S_1, \ldots, S_{n-1}$ — это случайные величины, которые равномерно распределены на $\{0,1\}^\ell$, и $S_n = \bigoplus\limits_{i=0}^{n-1} S_i$. Заметим, что S_0 определяется однозначно по остальным S_i, однако для любой перестановки σ на $[n]$ элементах:

$$\Pr\left[S_0 = a \mid S_{\sigma(1)}, \ldots, S_{\sigma(n-1)}\right] = \Pr[S_0 = a],$$

и, следовательно, $\mathrm{H}(S_0 \mid S_{\sigma(1)}, \ldots, S_{\sigma(n-1)}) = \mathrm{H}(S_0)$.

Мы можем немного усложнить нашу задачу и потребовать, чтобы секрет могли открыть не все участники, а любой кворум из k участников. Пусть $(S_0, S_1, \ldots, S_n)$ — это набор случайных величин. Будем называть его **схемой разделения секрета с порогом** k, если:

1. $\mathrm{H}(S_0 \mid S_I) = 0$, где $|I| \geq k$;
2. $\mathrm{H}(S_0 \mid S_I) = \mathrm{H}(S_0)$, где $|I| < k$.

Схема Шамира. Будем считать, что секрет S_0 — это элемент некоторого конечного $\mathbb{F}_q$. Зафиксируем набор различных точек $x_1, \ldots, x_n \in \mathbb{F}_q$ (в частности это условие нам говорит, что $q \geq n$). И рассмотрим многочлен:

$$P(x) = \sum_{i=1}^{t-1} a_i x^i + S_0,$$

коэффициенты a_i которого выберем случайным образом.

Пусть i-ый игрок получает число $S_i = P(x_i)$. Тогда любые t игроков смогут интерполировать многочлен и узнать S_0. Докажем, что $\mathrm{H}(S_0 \mid S_I) = \mathrm{H}(S_0)$, где $|I| < t$.

Пусть мы знаем значения S_i, то есть мы знаем значение $P(x_i) = c_i$, где $i \in I$. Это даёт нам линейную систему уравнений на коэффициенты полинома P с $|I|$ уравнениями и t неизвестными. Заметим, что:

$$\Pr[S_0 = S \mid \{P(x_i) = c_i\}]$$

не зависит от значений c_i, поскольку все x_i различны, а следовательно все уравнения линейно независимы. Таким образом мы получаем, что:

$$\Pr[S_0 = S \mid \{P(x_i) = c_i\}] = \Pr[S_0 = S],$$

и следовательно

$$\mathrm{H}(S_0 \mid S_I) = \mathrm{H}(S_0 \mid P(x_i)) = \mathop{\mathbb{E}}_{c_i, i \in |I|}[\mathrm{H}(S_0 \mid \{P(x_i) = c_i\}_{i \in I})] = \mathrm{H}(S_0).$$

Структуры доступа. Естественный обобщением пороговой схемы является схема в которой заданы некоторые произвольные «авторизованные» множества, которые могут узнать секрет, то есть у на задан набор подмножеств $\Gamma \subseteq 2^{[n]}$, замкнутый вверх. Пусть $(S_0, S_1, \dots, S_n)$ — это набор случайных величин. Будем называть его **схемой разделения секрета для структуры доступа** Γ, если:

1. $\mathrm{H}(S_0 \mid S_I) = 0$, где $|I| \in \Gamma$;
2. $\mathrm{H}(S_0 \mid S_I) = \mathrm{H}(S_0)$, где $|I| \notin \Gamma$.

Идеальная схема разделения секрета — это совершенная схема разделения секрета с дополнительным требованием «экономности».

$$\text{для всех } i \in \{1, 2, \dots, n\},\ \mathrm{H}(S_i) \leq \mathrm{H}(S_0).$$

Лемма 4.5

Если участник i является «существенным» в структуре доступа Γ (т.е. существует такое $s \in \Gamma$, что $s \setminus \{i\} \notin \Gamma$), то $\mathrm{H}(S_i) \geq \mathrm{H}(S_0)$.

Доказательство оставим в качестве упражнения.

Замечание 4.6

Если S_0 — равномерное распределение, то схема Шамира является идеальной.

Теорема 4.7

Существует такая структура доступа Γ, что для любой схемы разделения секрета выполнено неравенство $\max\limits_i \mathrm{H}(S_i) \geq \frac{n}{\log n} \mathrm{H}(S_0)$.

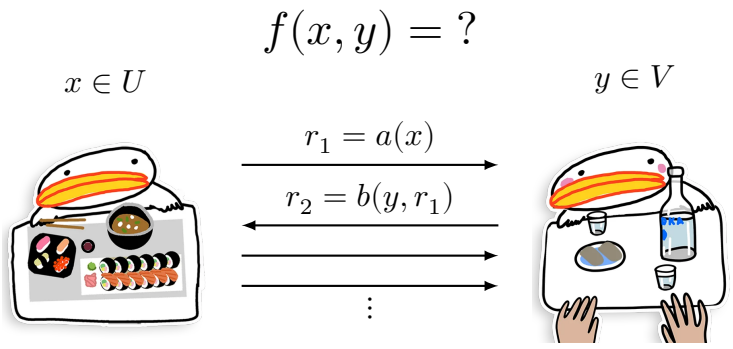
5. Коммуникационная сложность

Рассмотрим классическую задачу коммуникационной сложности. Пусть есть некоторая функция $f\colon X \times Y \to Z$ (причем не обязательно однозначная) и мы предполагаем, что все множества X, Y, Z конечны. Алиса и Боб хотят вместе посчитать значение этой функции (и оба узнать ответ), но Алиса знает значение только первого аргумета, а Боб только второго.

Чаще всего мы предполагаем, что Алиса и Боб не ограничены в вычислительных способностях.

В данном разделе мы рассмотрим, как классические коммуникационные задачи и техники, так и применения теории информаии в коммуникационной сложности.

Мы считаем, что, получив свои входы, Алиса и Боб начинают обмениваться друг с другом битами (в каждый момент времени только один из игроков посылает другому биты) до тех пор, пока однозначно не смогут определить ответ.



Определение 5.1

Коммуникационный протокол для функции $f\colon X \times Y \to Z$ — это корневое двоичное дерево, которое описывает совместное вычисление Алисой и Бобом функции f. В этом дереве:

- каждая внутренняя вершина v помечена меткой a или b, означающей очередь хода Алисы или Боба соответственно;
- для каждой вершины, помеченной a, определена функция $g_v\colon X \to \{0, 1\}$; аналогично, для каждой вершины v с пометкой b, определена функция $h_v\colon Y \to \{0, 1\}$;
- каждая внутренняя вершина имеет двух потомков, ребро к первому потомку помечено нулём, а ребро ко второму — единицей;
- каждый лист помечен значением из множества Z.

Пометки a или b, означают очередность хода Алисы или Боба соответственно. Функции g_v или h_v говорят какой бит нужно послать, если вычисление находится в вершине v. Таким образом, каждая пара входов (x, y) определяет путь от корня до листа в описанном двоичном дереве естественным образом. Будем говорить, что коммуникационный протокол **вычисляет** функцию f, если для всех пар $(x, y) \in X \times Y$ этот путь заканчивается в листе с пометкой $f(x, y)$.

Коммуникационной сложностью функции f называется наименьшая глубина протокола, вычисляющего функцию f. Будем обозначать её символом $\mathrm{D}(f)$.

Каждой функции f будем сопоставлять матрицу $X \times Y$, в которой в клетке (x_i, y_j) стоит значение $f(x_i, y_j)$.

Следующая лемма нам описывает комбинаторное свойство коммуникационных протоколов, которое нам позволяет доказывать нижние оценки.

Утверждение 5.2

Рассмотрим дерево протокола со входом из множества $X \times Y$. Рассмотрим в нём произвольную вершину u. Тогда все входы, из которых можно прийти в вершину u, образуют прямоугольник $R_u := X_u \times Y_u \subseteq X \times Y$.

Доказательство. Мы предъявим два способа доказать это утверждение.

Первый способ: пусть на входах (x_1, y_1) и (x_2, y_2) мы приходим в вершину u. Тогда нетрудно убедиться, что на входе (x_1, y_2) Алиса и Боб будут делать те же действия, что и на входах (x_1, y_1) и (x_2, y_2) соответственно. Отсюда видно, что входы, приводящие в вершину u, образуют прямоугольник $R_u = X_u \times Y_u \subseteq X \times Y$.

Второй способ: рассмотрим таблицу элементов $X \times Y$. После первого хода Алисы табличка делится пополам горизонтальной линией, так как при одних $x \in X$ Алиса посылает Бобу 1, а при других — 0. Если потом ход делает Боб, то каждый из двух получившихся прямоугольников делится своей вертикальной прямой, и так далее. В итоге мы получим разбиение $X \times Y$ на непересекающиеся прямоугольники, и каждый из этих прямоугольников соответствует листу в коммуникационном протоколе. □

Про прямоугольник R_u можно думать в следующим образом: если мы находимся в вершине протокола u, то нам необходимо решить задачу (то есть построить протокол) для всех входов из прямоугольника R_u. В частности этот подход можно рассмотреть, как комбинаторное определение протокола: бинарное дерево, в котором каждой вершине сопоставлен прямоугольник входов. И если вершины a, b являются потомками u, то $R_u \subseteq R_a \cup R_b$.

Рассмотрим величину $\chi_0(f)$, равную минимальному числу прямоугольников, которыми можно дизъюнктно покрыть нули в таблице. Аналогично определим $\chi_1(f)$. Тогда листьев в коммуникационном протоколе будет хотя бы $\chi_0(f) + \chi_1(f)$. Эти рассуждения дают следующую оценку:

$$\mathrm{D}(f) \geq \log(\chi_0(f) + \chi_1(f)).$$

Эта оценка не всегда точна. Рассмотрим такой пример разбиения таблицы $X \times Y$ на прямоугольники, где в центре находится прямоугольник из 1, а вокруг него расположены 4 прямоугольника из 0 (см. рис. 2). Заметим, что для этого разбиения не существует дерева протокола. Действительно, рассмотрим первое действие игроков. После него таблица должна поделиться на две части линией, проходящей через всю таблицу, но такого разреза не существует. Мы получили, что $\chi_0(f) + \chi_1(f) = 5$, хотя коммуникационного протокола с пятью листьями не существует.

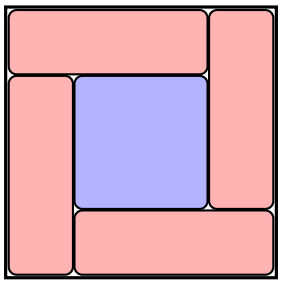


Рис. 2

5.1. Fooling Set

Пусть дана матрица некоторой функции. Выберем клетки этой таблицы $a_1, a_2, \ldots, a_n$ так, чтобы никакие две из них не могли оказаться в одном «одноцветном» прямоугольнике, то

есть состоящего только из нулей или только из единиц. Тогда для каждой клетки должен быть свой прямоугольник разбиения. Отсюда следует, что в дереве протокола должно быть хотя бы n вершин, а высота дерева хотя бы $\log n$. Этот метод называется **Fooling Set**.

Например, его можно применить для функции

$$\mathsf{EQ}(x,y) := \begin{cases} 1, & x = y, \\ 0, & x \neq y, \end{cases}$$

на строчках длины m мы получим диагональную матрицу $2^m \times 2^m$, и тогда в качестве точек a_i можно выбрать единицы на диагональной матрице. Понятно, что никакие две из них не находятся в одном одноцветном прямоугольнике. Значит высота коммуникационного дерева хотя бы $\log 2^m = m$.

Похожим образом мы можем доказать нижнюю оценку на функцию $\mathsf{Disj}(x,y)$ на подмножествах множества $[n]$, которая принимает значение 1 тогда и только тогда, когда $x \cap y = \emptyset$. Для всех множеств $S \in [2^n]$ рассмотрим ячейку матрицы $a_S := (S, \overline{S})$. Заметим, что для $S \neq S'$ ячейки a_S и $a_{S'}$ не могут лежать в одном прямоугольнике, так как иначе в этом прямоугольнике лежат ячейки $(S, \overline{S'})$, $(S', \overline{S})$, и в одной из них стоит 0. Из чего мы можем заключить, что высота коммуникационного дерева хотя бы $\log 2^n = n$.

5.2. Игры Карчмера–Вигдерсона

Определение 5.3

Формульной сложностью $L(f)$ формулы f будем называть минимальное возможное число листьев дерева, вычисляющего эту формулу (в вершинах дерева стоят булевы операции $\vee$, $\wedge$, а на некоторых рёбрах стоят $\neg$).

Теорема 5.4 [Шеннон]

Существует такая функция $f\colon \{0,1\}^n \to \{0,1\}$, что:

$$L(f) \geq \mathcal{O}\left(\frac{2^n}{n}\right).$$

Доказательство. Заметим, что количество функций $f\colon \{0,1\}^n \to \{0,1\}$ равно 2^{2^n}. Посмотрим на всевозможные формулы сложности не более, чем s. Каждое из них представляет собой двоичное дерево с не более, чем $2s-1$ вершинами, в которых написаны булевы операции, и не более, чем s листьями, в которых написаны переменные. Тогда в каждой вершине стоит один из $n+6$ символов (n переменных и 2 булевых символа с возможными отрицаниями перед ними), а количество деревьев с не более, чем $2s-1$ вершинами, по теореме Кэли не превосходит:

$$(2s-1)^{2s-3}(2s-1) \leq (2s)^{2s} \leq 2^{4s\log s}.$$

Отсюда следует, что количество деревьев сложности не более, чем s, не превосходит $2^{4s\log s + 2s\log(n+6)}$. Осталось убедиться, что при $s < \frac{2^n}{10n}$ количество деревьев сложности не более, чем s не превосходит количество функций $f\colon \{0,1\}^n \to \{0,1\}$. Действительно, при $s < \frac{2^n}{10n}$ верно, что $2^{4s\log s + 2s\log(n+6)} < 2^{2^n}$. А значит существует формула, сложность которой хотя бы $\frac{2^n}{10n}$. □

На данный момент «самая сложная» известная «явная», то есть для неё есть алгоритм, считающий её за полиномиальное время, функция f, для которой выполнено $L(f) \geq n^{3-\varepsilon}$.

Рассмотрим функцию $f\colon \{0,1\}^n \to \{0,1\}$. Для f можно рассмотреть следующую коммуникационную задачу KW_f: Алиса получает число $x \in f^{-1}(1)$, Боб — $y \in f^{-1}(0)$. Их цель —

найти хотя бы одну позицию i, в которой $x_i \neq y_i$. В случае, если таких битов несколько, то подойдет любой.

Теорема 5.5 [Карчмер–Вигдерсон]

Любую формулу для f можно переделать в коммуникационный протокол для KW_f с таким же деревом и обратно. В частности, минимальная глубина формулы для f равна коммуникационной сложности KW_f.

Доказательство. Функцию, которую считает гейт формулы u, будем обозначать f_u; аналогично, прямоугольник, соответствующий вершине протокола v, будем обозначать R_v.

Начнем с «простой» части и построим по формуле коммуникационный протокол. Пусть Алиса получила на вход строку $x \in f^{-1}(1)$, а Боб строку $y \in f^{-1}(0)$. Назовём гейт формулы u **хорошим**, если $f(x) \neq f(y)$.

Будем строить протокол, начиная с корня дерева формулы. По условию задачи, корень — это хорошая вершина, целью Алисы и Боба на каждом раунде будет являться поиск хорошего предка. И тогда, перемещаясь на каждом раунде в такого предка, Алиса и Боб найдут хороший лист, в котором и будет тот вход формулы, на котором строки Алисы и Боба различаются. Таким образом, для того, чтобы завершить доказательство, нам достаточно показать, как найти хорошего предка, передав не более одного бита. Рассмотрим текущую вершину u с предками a, b (НУО считаем, что $f_u(x) = 1 \wedge f_u(y) = 0$). У нас возможны два случая.

1. В вершине u написан значок $\wedge$. Тогда заметим, что $f_a(x) = f_b(x) = 1$, и при этом либо $f_a(y) = 0$, либо $f_b(y) = 0$. Таким образом, Боб может однозначно определить, какой из предков является хорошим, и сообщить это Алисе, передав один бит.

2. В вершине u написан значок $\vee$. Тогда заметим, что $f_a(y) = f_b(y) = 0$, и при этом либо $f_a(x) = 1$, либо $f_b(x) = 1$. Таким образом, Алиса может однозначно определить, какой из предков является хорошим, и сообщить это Бобу, передав один бит.

Теперь по протоколу построим формулу. По индукции, начиная с листьев протокола, для каждой вершины протокола v мы предъявим формулу для такой функции f_v, что $f_v(X_v) = 1$ и $f_v(Y_v) = 0$, где $R_v := X_v \times Y_v$.

Рассмотрим лист протокола ℓ и заметим, что прямоугольник $R_\ell := X_\ell \times Y_\ell$ одноцветный, поэтому все строки $x \in X_\ell$ отличаются от всех строк $y \in Y_\ell$ в какой-то фиксированной позиции i_ℓ, в частности это означает, что выполнен один из двух следующих случаев:

- для всех $x \in X_\ell, y \in Y_\ell$ бит $x_{i_\ell} = 1$ и $y_{i_\ell} = 0$, тогда мы можем определить $f_\ell := x_{i_\ell}$;
- для всех $x \in X_\ell, y \in Y_\ell$ бит $x_{i_\ell} = 0$ и $y_{i_\ell} = 1$, тогда мы можем определить $f_\ell := \neg x_{i_\ell}$.

Других случаев не существует, поскольку если найдутся такие $x, x' \in X$, что $x_{i_\ell} \neq x'_{i_\ell}$, то они одновременно не могут отличаться в позиции i_l ни от какого y (случай $y_{i_\ell} \neq y'_{i_\ell}$ аналогичен).

Построим теперь формулу для функции f_v, если у нас уже есть формулы для функций f_a и f_b, где a, b — потомки вершины v. Заметим, что прямоугольники R_a и R_b получены рассечением прямоугольника R_v на две части либо вертикальным, либо горизонтальным сечением. Рассмотрим эти случай отдельно и заметим, что:

- если сечение было горизонтальным, то $Y_a = Y_b = Y_v$ и $X_v = X_a \cup X_b$ и в таком случае нам подойдет формула $f_v := f_a \vee f_b$;
- если сечение было вертикальным, то $X_a = X_b = X_v$ и $Y_v = Y_a \cup Y_b$ и в таком случае нам подойдет формула $f_v := f_a \wedge f_b$.

□

5.3. Применения теории информации в коммуникационной сложности

Начнем с урезанной коммуникационной модели. Определим функцию индексирования:

$$\mathsf{Ind}\colon [n] \times \{0,1\}^n \to \{0,1\}, \qquad \mathsf{Ind}(x,y) = y_x,$$

где y_x — x-ый бит числа y. Рассмотрим следующую коммуникационную задачу, в которой Алиса получает на вход число $x \in [n]$, а Боб получает $y \in \{0,1\}^n$. Им нужно найти $\mathsf{Ind}(x,y)$. Нетрудно показать, что коммуникационная сложность этой задачи равна $\mathcal{O}(\log n)$, так как Алиса может послать x, и по x Боб может сказать бит, стоящий в x-ой позиции.

Усложним задачу. Пусть биты может посылать только Боб. Тогда коммуникационная сложность этой задачи становится $\mathcal{O}(n)$, так как в итоге Боб должен послать информацию про все биты своего числа. Иначе у Алисы может быть бит x, про который она ничего не знает.

Будем теперь считать, что Алиса и Боб получают входы согласно равномерному распределению среди всех возможных входов. Посылать информацию может только Боб. Но теперь Бобу с Алисой разрешено делать ошибку $\varepsilon := \frac{1}{2} - \delta$, то есть не больше, чем на $\varepsilon\,|X \times Y|$ входах можно сделать ошибку. Докажем, что коммуникационная сложность этой задачи равна $\mathcal{O}(\delta^2 n)$.

Рассмотрим коммуникационный протокол π, решающий нашу задачу. Заметим, что $\mathrm{D}^1(\pi) \geq \log|M|$, где M — множество листьев, а D^1 — коммуникационная сложность задачи, в которой только Боб может посылать биты. Рассмотрим энтропию $\mathrm{H}(M)$ распределения на листьях, которая получается естественным способом из распределения на входах. Тогда нетрудно получить следующие неравенства:

$$\log|M| \geq \mathrm{H}(M) \geq I(M\colon y).$$

По *chain rule* мы получаем, что:

$$\begin{aligned}
I(M\colon y) &= \sum_i I(M\colon y_i \mid y_{<i}) \\
&= \sum_i \mathrm{H}(y_i \mid y_{<i}) - \mathrm{H}(y_i \mid M, y_{\leq i}) \\
&= \sum_i \mathrm{H}(y_i) - \mathrm{H}(y_i \mid M, y_{<i}) && \text{так как все } y_i \text{ независимы} \\
&\geq \sum_i \mathrm{H}(y_i) - \mathrm{H}(y_i \mid M) \\
&\geq \sum_i I(M\colon y_i).
\end{aligned}$$

Для оценки $I(M\colon y_i)$ нам потребуется вспомогательная величина. Пусть r_i^m — вероятность ошибки, при условии того, что Боб послал сообщение m и $x = i$. Из определения r_i^m следует, что $\mathop{\mathbb{E}}_{i,m}[r_i^m] \leq \frac{1}{2} - \delta$.

Интуиция, скрывающаяся за следующими действиями такова: если $I(M\colon y_i)$ малая величина, то сообщение Боба «почти ничего» не сообщает об i-ом бите и, как следствие, если у Алисы $x = i$, мы получим ошибку с вероятностью примерно $\frac{1}{2}$. Попробуем формализовать

эту стратегию.

$$\begin{aligned} I(M\colon y_i) &= \mathrm{H}(y_i) - \mathrm{H}(y_i \mid M) \\ &= 1 - \mathrm{H}(y_i \mid M) && \text{поскольку распределение равномерное} \\ &= 1 - \mathop{\mathbb{E}}_m[\mathrm{H}(y_i \mid M = m)] \\ &= 1 - \mathop{\mathbb{E}}_m[\mathrm{H}(y_i \mid M = m, x = i)] && y_i \text{ и } M \text{ независимы относительно } x \\ &= 1 - \mathop{\mathbb{E}}_m[\mathrm{H}(r_i^m)]. \end{aligned}$$

Теперь попробуем оценить всю сумму:

$$\begin{aligned} \sum_i I(M\colon y_i) &= \sum_i 1 - \mathop{\mathbb{E}}_m[\mathrm{H}(r_i^m)] \geq && \text{неравенство Йенсена} \\ &\geq \sum_i 1 - \mathrm{H}\left(\mathop{\mathbb{E}}_m[r_i^m]\right) \\ &= n - n\sum_i \frac{1}{n}\,\mathrm{H}\left(\mathop{\mathbb{E}}_m[r_i^m]\right) \geq && \text{неравенство Йенсена} \\ &\geq n - n\,\mathrm{H}\left(\mathop{\mathbb{E}}_{i,m}[r_i^m]\right) \\ &\geq n\left(1 - \mathrm{H}\left(\frac{1}{2} - \delta\right)\right) \\ &= \Omega(\delta^2 n). \end{aligned}$$

Похожих идей будем придерживаться и в случае обычных коммуникационных протоколов.

Определение 5.6

Пусть $f\colon X \times Y \to Z$ и μ — распределение на $X \times Y$. Заметим, что для любого коммуникационного протокола Π для функции f распределение μ индуцирует распределение на листьях данного протокола естественным образом. **Внешней информационной стоимостью** (или **внешним информационным разглашением**) протокола Π по распределению μ будем называть величину:

$$\mathrm{IC}^{\mathrm{ext}}_\mu(\Pi) := I(\Pi(X, Y)\colon X, Y).$$

Также определим внешнюю информационную сложность самой функции $\mathrm{IC}^{\mathrm{ext}}_\mu(f) := \min\limits_{\Pi} \mathrm{IC}^{\mathrm{ext}}_\mu(\Pi)$.

По аналогии мы может рассмотреть и **внутреннее информационное разглашение** (или **внутренюю информационную стоимость**).

$$\mathrm{IC}^{\mathrm{int}}_\mu(\Pi) := I(\Pi(X, Y)\colon X \mid Y) + I(\Pi(X, Y)\colon Y \mid X).$$

Следующая теорема на дает связь между коммуникационной сложностью и новой мерой сложности.

Теорема 5.7

Пусть π — протокол некоторой коммуникационной задачи с мерой μ на входах. Тогда:

$$\mathrm{D}(\pi) \geq \mathrm{IC}^{\mathrm{ext}}_\mu(\pi) \geq \mathrm{IC}^{\mathrm{int}}_\mu(\pi).$$

Доказательство. $\mathrm{D}(\pi) \geq \log(M) \geq \mathrm{H}(\pi) \geq I(\pi\colon X, Y)$, где $\mathrm{H}(\pi)$ — энтропия от распределения на листьях. Откуда следует первое неравенство.

Теперь докажем второе: $I(\pi\colon X, Y) \geq I(\pi\colon X \mid Y) + I(\pi\colon Y \mid X)$. По chain rule мы получаем, что $I(\pi\colon X, Y) = \sum\limits_i I(\pi_i\colon X, Y \mid \pi_{<i})$, где π_i — это случайная величина, соответствующая i-ому биту, переданному в протоколе. Мы хотим показать, что:

$$I(\pi_i\colon X, Y \mid \pi_{<i}) \geq I(\pi_i\colon X \mid Y, \pi_{<i}) + I(\pi_i\colon Y \mid X, \pi_{<i}),$$

Для доказательства заметим, что для любого $m \in \{0,1\}^{i-1}$:

$$I(\pi_i\colon X, Y \mid \pi_{<i} = m) \geq I(\pi_i\colon X \mid Y, \pi_{<i} = m)$$
$$I(\pi_i\colon X, Y \mid \pi_{<i} = m) \geq I(\pi_i\colon Y \mid X, \pi_{<i} = m)$$

Так как по m мы знаем в какой вершине протокола мы находимся после i шагов, то в этой вершине либо Алиса либо Боб посылают бит и π_i будет определяться однозначно либо по X, либо по Y. Откуда следует, что одна из величин $I(\pi_i\colon X \mid Y, \pi_{<i} = m)$, $I(\pi_i\colon Y \mid X, \pi_{<i} = m)$ равна 0. А поскольку $I(\pi_i\colon X, Y \mid \pi_{<i} = m)$ не меньше второй величины, то мы получаем, что:

$$I(\pi_i\colon X, Y \mid \pi_{<i} = m) \geq I(\pi_i\colon X \mid Y, \pi_{<i} = m) + I(\pi_i\colon Y \mid X, \pi_{<i} = m),$$

откуда требуемое неравенство получается усреднением m по мере μ.

По chain rule получаем, что:

$$\sum I(\pi_i\colon X \mid Y, \pi_{<i}) + I(\pi_i\colon Y \mid X, \pi_{<i}) = I(\pi\colon X \mid Y) + I(\pi\colon Y \mid X).$$

Окуда следует второе неравенство из условия теоремы. □

При этом для любого протокола π существует такая мера μ, что $\log|\pi| = \mathrm{IC}^{\mathrm{ext}}_{\mu}(\pi)$, где L — число листье в протоколе.

Теорема 5.8 [Храпченко]

$L(\oplus_n) \geq \mathcal{O}(n^2)$, где функция $\oplus_n$ равна чётности количества единиц в записи числа длины n.

Доказательство. Рассмотрим $\mathrm{KW}_{\oplus_n}$. Покажем, что $\mathrm{IC}^{\mathrm{int}}_{\mu}(\mathrm{KW}_{\oplus_n}) \geq 2\log n$, откуда будет следовать, что $\mathrm{D}(\oplus_n) \geq 2\log n$ и по теореме 5.5 мы получим $L(\oplus_n) \geq n^2$.

Определим распределение μ: равномерное распределение на парах $(x, x \oplus e_i)$, по всем $x \in \oplus_n^{-1}(1)$ (по всем x, в записи которых нечётное число единиц). Тогда:

$$I(\pi\colon Y \mid X) = I(\pi\colon x \oplus e_I \mid X) = I(\pi\colon e_I \mid X) = \mathrm{H}(e_I \mid X) - \mathrm{H}(\pi \mid X, e_I)$$
$$= \mathrm{H}(e_I \mid X) = \mathrm{H}(e_I) = \log n.$$

Аналогично со вторым слагаемым. □

Данная оценка является точной.

6. Колмогоровская сложность

Определение 6.1

Пусть $F\colon \{0,1\}^* \to \{0,1\}^*$ — вычислимая функция. **Сложностью описания** x относительно функции F мы будем называть:

$$\mathrm{K}_F(x) = \min\{|p| : F(p) = x\}.$$

Если такого p не найдётся, то $\mathrm{K}_F(x) = +\infty$.

Будем говорить, что способ описания F **не хуже** способа описания G, и обозначать $F \prec G$ если существует такая константа c_{FG}, что для любого x выполнено:

$$\mathrm{K}_F(x) \leq \mathrm{K}_G(x) + c_{FG}.$$

Будем называть способ описания **оптимальным**, если он не хуже любого другого способа описания.

Теорема 6.2

Существует оптимальный способ описания.

Доказательство. Доказательство оставим в качестве упражнения. □

Если F, G — оптимальные способы описания, то $|\mathrm{K}_F(x) - \mathrm{K}_G(x)| \leq c_{FG}$ для всех x и некоторой константы c_{FG}.

Определение 6.3

Будем называть **колмогоровской сложностью** x число $\mathrm{K}(x) := \mathrm{K}_u(x)$, где u — какой-то оптимальный способ описания.

Мы не уточнили, какой из оптимальных способов описания рассматриваем, как следствие это означает, что мы определили колмогоровскую сложность с точностью до аддитивной константы. Поэтому говорить о сложности одной строки не имеет смысла (действительно, если взять конкретную строку w, то у нас существует алгоритм, в тексте которого уже присутствует эта строка w, и он выписывает её на пустом входе). Чтобы данное определение обрело смысл, мы будем иметь ввиду, что у нас есть семейство строк, параметризованное каким-либо параметром (чаще всего n).

Рассмотрим некоторые простые свойства колмогоровской сложности.

1. $K(x) \leq |x| + c$;
2. $K(xx) \leq |x| + c$;
3. если в слове x длины n не более np единиц, то

$$\mathrm{K}(x) \leq \mathrm{H}(p)n + \mathcal{O}(1),$$

где $\mathrm{H}(p)$ — энтропия нечестной монетки, выпадающей орлом с вероятностью p, где $0 \leq p \leq 1$.

Первое и второе утверждения следуют и того, что существует алгоритм, копирующий свой вход на выходную ленту, а также алгоритм, который делает это дважды.

Рассмотрим множество S, состоящее их всех слов длины n, в которых не более pn единиц. Заметим, что $x \in S$, и его можно описать при помощи номера в этом множестве (подойдёт любая нумерация элементов). Осталось заметить, что:

$$|S| \leq \sum_{k=1}^{np} \binom{n}{k} \leq 2^{\mathrm{H}(p)n},$$

где последнее неравенство следует из утверждения 2.6.

Теорема 6.4

Рассмотрим такую всюду определенную функцию $M\colon \{0,1\}^* \to \mathbb{N}$, что для любого $x \in \{0,1\}^*$ верно $M(x) \leq \mathrm{K}(x)$. Если для любой константы c существует такое $w \in \{0,1\}^*$, что $M(w) > c$, то M — невычислима.

Хотим провести рассуждения, аналогичные рассуждениям в *парадоксе Бэрри*: рассмотрим выражение «Наименьшее натуральное число, которое нельзя описать менее чем одиннадцатью русскими словами». Поскольку слов конечное число, существует конечное множество фраз из менее чем одиннадцати слов, и, следовательно, конечное подмножество натуральных чисел, определяемых фразой из одиннадцати слов. Однако множество натуральных чисел бесконечно, следовательно, существуют числа, которые нельзя определить фразой из менее чем одиннадцати слов. Среди них, существует наименьшее натуральное число (наименьшее число можно выбрать из любого подмножества натуральных чисел), «не описываемое менее чем одиннадцатью словами». Но именно это число определяется приведённой выше фразой, и в ней менее одиннадцати слов, а значит, оно не может являться искомым наименьшим числом и не может описываться данной фразой. Возникает парадокс: должно существовать число, описываемое данной фразой, но поскольку выражение само себе противоречит, не может существовать числа, им описываемого.

Доказательство. Рассмотрим первое слово x_c, для которого выполняется условие $M(x_c) \geq c$ для некоторой константы c. Определим функцию $F(\overline{c}) = x_c$, где $\overline{c}$ — битовая запись числа c. Заметим, что если M вычислима, то и функция F будет вычислимой. Но тогда $\mathrm{K}(x_c) \leq \log c + c_0$, поскольку $|\overline{c}| = \log c$, и мы получаем противоречие, так как $M(x_c) = c > \log c = \mathrm{K}(x_c)$. □

Следствие 6.5

Любой оптимальный способ определим не всюду.

Теорема 6.6

99% слов $x \in \{0,1\}^n$ имеют сложность $n - \mathcal{O}(1)$.

Доказательство. Зафиксируем какой-то оптимальный способ описания U. Тогда число описаний длины не более $n - c$ не превосходит 2^{n-c+1}. Тогда у хотя бы $2^n - 2^{n-c+1}$ длина кратчайшего описания не менее $n - c + 1$, откуда нетрудно получить утверждение теоремы. □

6.1. Условная колмогоровская сложность

Рассмотрим вычислимую функцию $F(p, y) = x$. Будем говорить, что **сложность описания x относительно функции F при условии y** равна

$$\mathrm{K}_F(x \mid y) = \min\{|p| : F(p, y) = x\}.$$

Теорема 6.7

Для любого y существует оптимальный способ описания.

Определение 6.8

Определим колмогоровскую сложность $\mathrm{K}(x \mid y) = \mathrm{K}_u(x \mid y)$ для некоторого оптимального способа описания u при фиксированном y.

Для условной колмогоровской сложности выполнены естественные свойства:

- $\mathrm{K}(x \mid y) < \mathrm{K}(x) + \mathcal{O}(1)$;
- $\mathrm{K}(f(x) \mid x) = \mathcal{O}(1)$ для любой вычислимой функции f.

Определение 6.9

Определим колмогоровскую сложность пары $\mathrm{K}(x, y) = \mathrm{K}(\langle x, y\rangle)$, где $\langle x, y\rangle$ — фиксированная кодировка.

Для энтропии мы знаем, что $\mathrm{H}(x, y) = \mathrm{H}(x) + \mathrm{H}(y \mid x)$. Хочется доказать аналогичное утверждение про колмогоровскую сложность. Равенство для колмогоровской сложности не всегда верно. Также неверно следующее неравенство:

$$\mathrm{K}(x, y) \leq \mathrm{K}(x) + \mathrm{K}(y \mid x) + \mathcal{O}(1).$$

Это можно видеть из следующего примера.

Пример 6.10

Рассмотрим пары (x, y), для которых $|x| + |y| = n$. Таких пар $n \cdot 2^n$. Аналогично доказательству теоремы 6.6 найдётся пара сложности $n + \log n - \mathcal{O}(1)$. Тогда для неё мы получим, что

$$\mathrm{K}(x, y) = \mathrm{K}(x) + \mathrm{K}(y \mid x) + \log n + \mathcal{O}(1).$$

Мы можем чуть-чуть поправить неравенство, чтобы оно стало верным.

Теорема 6.11

$$\mathrm{K}(x, y) \leq \mathrm{K}(x) + \mathrm{K}(y \mid x) + \mathcal{O}(\log \mathrm{K}(x, y)).$$

Доказательство. Пусть $n = n_1 n_2 \ldots n_k$ — длина описания x. Тогда пару $\langle x, y\rangle$ можно закодировать числом:

$$n_1 n_1 n_2 n_2 \ldots n_k n_k 01\{\text{описание } x\}\{\text{описание } y \text{ при известном } x\}.$$

Что дает нам описание (x, y) длины $\mathrm{K}(x) + \mathrm{K}(y \mid x) + \mathcal{O}(\log \mathrm{K}(x, y))$. □

Теорема 6.12 [Колмогоров–Левин]

Имеет место равенство

$$\mathrm{K}(x, y) = \mathrm{K}(x) + \mathrm{K}(y \mid x) + \mathcal{O}(\log \mathrm{K}(x, y)).$$

Доказательство. Неравенство в одну сторону мы показали. Докажем неравенство в другую сторону.

Положим $n := \mathrm{K}(x, y)$. Определим множество $S := \{(a, b) \mid \mathrm{K}(a, b) \leq n\}$. Будем рассматривать сечение S_x. Пусть $\log |S_x| = m$. Тогда для описания y по x нам нужно перечислить элементы S_x и найти y. Для этого нам нужно знать n, это потребует $\mathcal{O}(\log n)$ битов, и нужно уметь перечислять элементы S_x, для этого нам потребуется m битов. Тогда:

$$\mathrm{K}(y \mid x) \leq m + \mathcal{O}(\log n).$$

Для описания x рассмотрим все такие t, что $\log |S_t| \geq m$, и выберем из них x. Для этого нам опять потребуется $\log n$ битов, для хранения n, и $n - m$ битов для перечисления t, так как таких t не больше, чем $\frac{|S|}{2^m} \leq 2^{n+1-m}$, так как $|S| \leq 2^{n+1}$. Откуда:

$$\mathrm{K}(x) \leq n - m + \mathcal{O}(\log n).$$

Тогда получаем:

$$\mathrm{K}(x) + \mathrm{K}(y \mid x) \leq n + \mathcal{O}(\log n) = \mathrm{K}(x, y) + \mathcal{O}(\log \mathrm{K}(x, y))$$

□

Можно определить $I_{\mathrm{K}}(x\colon y) = \mathrm{K}(y) - \mathrm{K}(y \mid x)$. Также верно неравенство

$$|I_{\mathrm{K}}(x\colon y) - I_{\mathrm{K}}(y\colon x)| \leq \log \mathrm{K}(x, y) + \mathcal{O}(1).$$

От логарифма избавиться не получится, можно рассмотреть такой $x \in \{0, 1\}^n$, что $\mathrm{K}(x \mid n) = n - \mathcal{O}(1)$. Тогда $I_{\mathrm{K}}(n\colon x) = \mathcal{O}(1)$, а $I_{\mathrm{K}}(x\colon n) = \log n$.

7. Применения колмогоровской сложности

Колмогоровская сложность применяется в задачах, связанных с попытками доказать или опровергнуть **NP**-полноту задачи MCSP, в которой по таблице истинности функции $f\colon \{0, 1\}^n \to \{0, 1\}$ и числу $s < 2^n$ необходимо определить, существует ли булева схема размера s, которая считает данную функцию. Нетрудно понять, что эта задача принадлежит классу **NP**. Но не доказано, является ли эта задача **NP**-полной.

Рассмотрим более простую задачу. Обозначим за L_k множество языков, принимаемых детерминированными автоматами с одной лентой и k головками на ней, причём головки едут только в одну сторону.

Утверждение 7.1

$L_{k+1} \not\subseteq L_k$.

Доказательство. Рассмотрим язык L, состоящий из строк вида:

$$w_1 \# w_2 \# \ldots \# w_{m-1} \# w_m \# w_m \# w_{m-1} \# \ldots \# w_2 \# w_1,$$

где m — фиксировано, а w_i — произвольные слова. Покажем, что если $m > \frac{k(k-1)}{2}$, то язык L не распознаётся автоматом с k головками, а если это не так, то распознаётся.

В каждом слове языка L есть по два слова w_i. Будем называть их «левым» и «правым» соответственно. Пусть $m \leq \frac{k(k-1)}{2}$. Опишем автомат с k головками, который распознаёт язык L. Сначала все k головок расположены на началах левых слов $w_1, w_2, \ldots, w_k$. Следующим действием последняя головка, которая стоит в начале w_k, доходит до начала правого слова w_{k-1}. Далее проверяем равенство левых и правых слов w_i, где $1 \leq i \leq k - 1$. Тем самым, «потратив» одну головку мы проверили равенство $k - 1$ слова сначала и с конца. Далее

повторяем алгоритм для $k-1$ головки и слова, без первых и последних $k-1$ слов w_i. Тем самым мы проверим $(k-1)+(k-2)+\ldots+1=\frac{k(k-1)}{2}$ слов.

Пусть теперь $m>\frac{k(k-1)}{2}$. Посмотрим на левый и правый блоки с номером ℓ (слово w_ℓ). Пусть в какой-то момент головки i и j стоят на разных копиях слова w_l. Тогда эта пара головок (i,j) ни в какой другой момент времени не будет находиться в одинаковых словах w_r, при $r\neq\ell$, так как головки движутся только вправо. Тогда, так как $m>\frac{k(k-1)}{2}$, то для какого-то слова w_ℓ верно, что в каждый момент времени только в одной копии w_ℓ может находиться головка.

Построим протокол, который будет запоминать состояния автомата и позиции головок для каждого такого момента, что либо какая-то головка оказалась в слове w_ℓ, либо какая-то головка покинула слово w_ℓ. Попробуем восстановить w_ℓ по протоколу. Подставим x на место w_ℓ. Если автомат выдал такой же протокол работы, то x потенциально подходит. Предположим, что нашлось 2 подходящих слова x и y, для которых протокол работает правильно. Тогда рассмотрим слово $w_{x,y}$, равное слову w, в котором левое слово w_ℓ заменили на x, а правое слово w_ℓ заменили на y. Заметим, что для слов w и $w_{x,y}$ алгоритм работает одинаково. А значит $w_{x,y}$ лежит в нашем языке, что не правда. Откуда получаем противоречие, а значит по протоколу мы однозначно восстанавливаем w_ℓ.

Пусть длина слова w равна n, и в нём зафиксированы слова $w_i, i\neq\ell$. Тогда размер протокола для w равен $\mathcal{O}\left(k^3\log n\right)$, где $k-$ число головок. Осталось заметить, что символов для кодирования строки w, с фиксированными $w_i, i\neq\ell$, не меньше, чем $\sum\limits_{i\neq\ell}|w_i|+\mathcal{O}\left(k^3\log n\right)$. Заметим, что существуют такие w_i, что $|w_i|=s$ для любого i и $K(w_1\#\ldots\#w_m\#w_m\#\ldots\#w_1)\geq ms-c$. Но из предыдущих рассуждений мы получили, что колмогоровская сложность w равна $(m-1)s+\mathcal{O}\left(\log s\right)$, так как $\sum\limits_{i\neq l}|w_i|=(m-1)s$. Тогда при больших s мы получаем противоречие. А значит язык L не лежит в L_k, теорема доказана. □

8. Случайные по Мартин-Лёфу

Рассмотрим бесконечную строчку $w=x_1x_2x_3\ldots$ Хотим узнать, случайна ли эта строчка, или она получена каким-то алгоритмом. Предположим, что мы знаем биты $x_1,x_2,\ldots,x_{n-1}$. Можно ли узнать бит x_n? Если бы можно было узнать все биты x_n по предыдущим, то понятно, что строка не случайная. Но как часто можно узнавать следующий бит по предыдущим в случайной строке?

Хотелось бы сказать, что строка случайная, если существует такая константа c, что для любого n верно $\mathrm{K}(x_{\leq n})\geq n-c$ (то есть когда колмогоровская сложность достаточно большая). Оказывается, что не существует такой последовательности x_i, для которой это неравенство выполнено.

Теорема 8.1

Для любой последовательности $x_1x_2x_3\ldots$ и числа n_0 найдётся такое $n>n_0$, что $\mathrm{K}(x_{\leq n})\leq n-\mathcal{O}\left(\log n\right)$. В частности, для любой константы c найдётся такое n, что $\mathrm{K}(x_{\leq n})\leq n-c$.

Доказательство. Рассмотрим любую последовательность $x_1x_2x_3\ldots$ и её префиксы $x_1\ldots x_kx_{k+1}\ldots x_{k+m}$, где $k+m=n$. Рассмотрим префиксы длины n, которые разбиваются на две части, $x_1\ldots x_k$ и $x_{k+1}\ldots x_n$, причём $1x_1x_2\ldots x_k=m$ (не все n подходят). Тогда $\mathrm{K}(x_{\leq m+k})\leq m+\mathcal{O}\left(1\right)$, так как зная последние m символов префикса длины n мы можем восстановить весь префикс. Но длина всего префикса равна $n=m+\log m$, откуда получаем, что $\mathrm{K}(x_{\leq n})\leq n-\mathcal{O}\left(\log n\right)$. Осталось заметить, что n можно выбрать сколь угодно большое,

что и доказывает теорему. □

Определение 8.2

Функцию F будем называть **беспрефиксной**, если из того, что F определена на x и y следует, что x не является префиксом y, и наоборот.

Для беспрефиксных функций можно определить понятие колмогоровской сложности так же, как и для обычных функций. Мы будем обозначать её через $\mathrm{KP}(x)$.

Определение 8.3

Строку w будем называть **случайной**, если существует такая константа c, что $\mathrm{KP}(w_{\leq n}) \geq n - c$ для всех $n \in \mathbb{N}$.

В определении колмогоровской сложности мы пользовались тем, что существует оптимальный способ описания для беспрефиксных функций. Докажем это в следующей теореме.

Теорема 8.4

Существует оптимальный беспрефиксный способ описания.

Для доказатетельства данной теоремы нам понадобится вспомогательное утверждение.

Теорема 8.5

Существует такой алгоритм A, что для любой вычислимой функции F выполняются условия:

- если F — беспрефиксная, то $A(F) = F$;
- если F — не беспрефиксная, то $A(F) = F'$, где F' — беспрефиксная.

Доказательство. Пусть на вход дана функция F, и мы хотим посчитать $F(x)$. Запустис F параллельно на всех входах (стандартная операция из теории вычислимости). Если в какой-то момент, до того, как мы посчитали $F(x)$ оказалось, что алгоритм подсчета F остановился двух строках y и yz для некоторых y и z, то зацикливается. Иначе выдаем значение $F(x)$.

Заметим, что если алгоритму A дали на вход беспрефиксцю функцию, то алгоритм A выдает ее же, и A всегда выдаёт какую-то беспрефиксную функцию. □

Доказательство. В качестве оптимального способа описания мы можем использовать $U'(x, y) := A(x(y))$, где (x, y) задано беспрефиксным образом, например все биты x удваиваются, затем идет 01, а затем кодировка y. □

Заметим, что $\mathrm{KP}(x) \leq \mathrm{K}(x) + 2\log \mathrm{K}(x) + \mathcal{O}(1)$. Чтобы доказать это неравенство применим кодировку: $p \to a_1 a_1 a_2 a_2 \ldots a_k a_k 01 p$, где p — описание x и $|p| = a_1 a_2 \ldots a_k$. Тогда $k \leq \log \mathrm{K}(x)$ и $\mathrm{KP}(x) \leq \mathrm{K}(x) + 2\log \mathrm{K}(x) + \mathcal{O}(1)$.

Теорема 8.6

Мера Бернулли неслучайных последовательностей равна 0.

Доказательство. Рассмотрим некоторую неслучайную последовательность $x_1 x_2 \ldots$. Тогда так как она не случайна, то для любого c существуют $n_1, n_2, \ldots, n_l, \ldots$, что $\mathrm{KP}(x_{\leq n_i}) \leq n_i - c$. Тогда

$$\sum_{\{x | \mathrm{KP}(x) \leq n-c, |x|=n\}} 2^{-n} \leq \sum_x 2^{-\mathrm{KP}(x)-c} \leq 2^{-c} \cdot \sum_x 2^{-\mathrm{KP}(x)} \leq 2^{-c}.$$

Последнее неравенство верно, так как код префиксный, и можно применить неравенство Крафта–Макмиллана. Также если A_i — множество строк с плохими префиксами $x_1x_2 \ldots x_i$, то мера Бернулли неслучайных последовательностей

$$M\left(\bigcup A_i\right) \leq \sum_{\{x|\mathrm{KP}(x)\leq n-c\}} 2^{-n} < 2^{-c}.$$

Осталось устремить c к бесконечности, и получить требуемое. □

Теорема 8.7 [Закон больших чисел в форме Харди–Литлвуда]

Для почти всех последовательностей $x := x_1x_2 \ldots x_n \ldots$ (с вероятностью 1) выполнено:

$$\left|\frac{x_1 + x_2 + \cdots + x_n}{n} - \frac{1}{2}\right| = \mathcal{O}\left(\sqrt{\frac{\log n}{n}}\right).$$

Доказательство. Пусть в $x_1 \ldots x_n$ всего $p_n \cdot n$ единиц и $(1 - p_n) \cdot n$ нулей.

$$\mathrm{KP}(x_1 \ldots x_n) \leq \mathrm{K}(x_1 \ldots x_n) + \mathcal{O}(\log n) \leq \mathrm{H}(p) \cdot n + \mathcal{O}(\log n).$$

Пусть $p := \frac{1}{2} + \delta_n$. Разложим $\mathrm{H}(p)$ в ряд в окрестности $\frac{1}{2}$:

$$\mathrm{H}(1/2 + \delta_n) \cdot n = (1 - c_H \cdot \delta_n^2 + o(\delta_n^2)) \cdot n \leq (1 - c'_H \cdot \delta_n^2) \cdot n.$$

Таким образом для случайной последовательности (т.е. с вероятностью 1):

$$n - c \leq \mathrm{KP}(x_1 \ldots x_n) \leq n - c'_H \cdot \delta_n^2 \cdot n + \mathcal{O}(\log n).$$

Получаем, что $\delta_n^2 \leq \mathcal{O}\left(\frac{\log n}{n}\right)$. □

Список литературы

[BGL10] Olaf Beyersdorff, Nicola Galesi, and Massimo Lauria. “A lower bound for the pigeonhole principle in tree-like Resolution by asymmetric Prover-Delayer games.” In: *Inf. Process. Lett.* 110.23 (2010), pp. 1074–1077. DOI: 10.1016/j.ipl.2010.09.007. URL: https://doi.org/10.1016/j.ipl.2010.09.007.

[KR11] Swastik Kopparty and Benjamin Rossman. “The homomorphism domination exponent.” In: *Eur. J. Comb.* 32.7 (2011), pp. 1097–1114. DOI: 10.1016/j.ejc.2011.03.009. URL: https://doi.org/10.1016/j.ejc.2011.03.009.